

Situationsbewusste Informationsdienste für das arbeitsbegleitende Lernen

Zur Erlangung des akademischen Grades eines
Doktors der Ingenieurwissenschaften
der Fakultät für Informatik der Universität Fridericiana zu Karlsruhe (TH)
genehmigte Dissertation
von

Andreas Schmidt
aus Pforzheim

Tag der mündlichen Prüfung: 11.02.2009

Erster Gutachter: Prof. Dr. Peter C. Lockemann
Zweiter Gutachter: Prof. Dr. Johannes S. Magenheimer

Kurzzusammenfassung

Motivation und Zielsetzung

Mit der steigenden Bedeutung des lebenslangen Lernens tritt immer mehr das »Lernen bei Bedarf« statt eines »Lernens auf Vorrat« in den Vordergrund. Dieses kann aber nur dann stattfinden, wenn Lernen und Arbeiten als miteinander verwobene Aktivitäten verstanden werden. Dies ist allerdings trotz zahlreicher Versuche des Einsatzes von Informationstechnologien in Form von E-Learning, Wissensmanagement u.a. bislang noch nicht zufriedenstellend gelungen. Das Kernproblem liegt darin, dass sie zu wenig die Arbeitssituation berücksichtigen, in der sie benutzt werden. Was also benötigt wird, sind situationsbewusste lernunterstützende Dienste.

In dieser Arbeit geht es darum, die Forderungen nach mehr Situationsbewusstsein aus der Problematik des arbeitsbegleitenden Lernens heraus zusammenzubringen mit der technischen Umsetzung von situationsbewussten Informationsdiensten. Ziel ist die Verbesserung des arbeitsbegleitenden Lernens durch Bereitstellung von Diensten, die unter Berücksichtigung der aktuellen Situation des Nutzers bzw. Lernenden die Lernprozesse unterstützen.

Dieses Gesamtziel gliedert sich in zwei wesentliche Teilziele:

- eine Untersuchung, wie durch Situationsbewusstsein von Diensten arbeitsbegleitendes Lernen unterstützt werden kann, welche Dienste im Grenzbereich der heutigen Wissensmanagement- und E-Learning-Ansätze hierfür benötigt werden und wie sie in eine Rahmenarchitektur einzuordnen sind, sowie die Realisierung exemplarischer Dienste
- und die Konzeption einer Infrastruktur zum Umgang mit Kontextinformationen als systemseitigem Abbild der Situation von Benutzern, und zwar basierend auf den Anforderungen, die sich aus den lernunterstützenden Diensten ergeben.

Lösungsansatz

Die der Arbeit zugrundeliegende Forschungsmethodik ist Design Research, deren wesentliche Ergebnisse ein konzeptuelles Vokabular, Modelle Methoden und deren Umsetzung

sind. Die Vorgehensweise zur Erzielung dieser Ergebnisse orientiert sich fünf Schritten: Entwickeln eines Problembewusstseins, Lösungsvorschlag, Lösungsentwicklung und Implementierung, Evaluation und Schlussfolgerung. Da der Hauptschwerpunkt der Arbeit auf der Lösungsentwicklung liegt, gliedert sich die Arbeit in fünf Hauptteile:

- Der erste Teil widmet sich als Beitrag zur Schaffung des Problembewusstseins dem arbeitsbegleitenden Lernen und dem notwendigen Verständnis für die Zusammenhänge und Probleme. Dabei geht in einem ersten Schritt um ein geeignetes Makromodell, das es erlaubt, das Themenfeld zu strukturieren, Beziehungen zwischen den individuellen Lernprozessen und den überindividuellen Abläufen zu schaffen und die Rolle der Situation im Großen zu verankern. Hierzu wird der Wissensreifungsprozess entwickelt.
- Im zweiten Teil als Kernlösungsvorschlag wird für einen der Problembereiche im Wissensreifungsprozess ein methodischer Rahmen für die Lernunterstützung entwickelt: das situationsgesteuerte Lernen.
- Dieser Rahmen bildet die Grundlage für ein technisches Rahmenwerk für die Lernunterstützung im dritten Teil; er liefert die Anforderungen für die Konzeption von lernunterstützenden situationsbewussten Diensten und die Entwicklung einer dienstorientierten Architektur, die eine situationsbewusste Lernunterstützung ermöglicht. Dabei geht es um die Identifikation von benötigten Diensten und die Spezifikation von Dienstschnittstellen und Interaktionsmustern. Auch das benötigte konzeptionelle Modell wird entwickelt, das die semantische Kohärenz der Dienste sicherstellt. Die technische Realisierung dieser situationsbewussten Dienste liefert die Anforderungen an die Kontextinfrastruktur.
- Als zweiter Baustein der Lösungsentwicklung kümmert sich der vierte Teil um eine Kontextverwaltungsinfrastruktur, die die Basis für die lernunterstützenden Dienste darstellt. Dabei werden als zentrale Herausforderungen die Unvollkommenheit und die Dynamik von Kontextinformationen identifiziert. Als Lösung werden ein Kontextmodell als Datenmodell sowie darauf aufbauende Implementierungstechniken vorgestellt, die effizient eine qualitativ hochwertige Sicht auf den aktuellen Kontext des Benutzers liefern.
- Im fünften Teil erfolgt schließlich die Implementierung von Demonstratoren und die Evaluierung ihres Einsatzes in realen Unternehmensumgebungen.

Wissenschaftlicher Beitrag

Wissensreifungsprozess als Makromodell für formelles und informelles Lernen

Der Wissensreifungsprozess liefert ein Modell, mit dem sich die komplexe Landschaft des betrieblichen und insbesondere des arbeitsbegleitenden Lernen strukturieren lässt. Es integriert sowohl den »Wissenstransfer« im Sinne des Wissensmanagements, als auch das »Lernen« im Sinne des geführten E-Learning erstmals in einem einheitlichen Modell, das die unterschiedlichen Fokusbereiche und Herangehensweisen erklärt: Durch den Wissensreifungsprozess wird sichtbar, wo die Brüche zwischen unterschiedlichen Konzeptionen zu finden sind.

Modell für situationsgesteuerte Lernprozesse

Die Methode des situationsgesteuerten Lernens kombiniert E-Learning-Ansätze aus dem Bereich der pädagogischen Agenten (proaktive Begleitung des Lernprozesses), adaptiven E-Learning-Ansätze und des kompetenzbasierten E-Learnings mit geschäftsprozessorientierter Informationsversorgung. Hierdurch können sowohl organisationale und persönliche sowie ergänzend soziale Aspekte Berücksichtigung finden. Im Gegensatz zu anderen kontext- oder situationsbewussten E-Learning-Methoden ist der Ansatz hierbei ganzheitlich und beschränkt sich nicht auf Teilaspekte.

Referenzontologie und -architektur für lernunterstützende Dienste

Mit der als Professional Learning Ontology"bezeichneten bislang einzigen integrativen Referenzontologie wurde die Grundlage geschaffen, die Bereiche Wissensmanagement, Personalentwicklung und E-Learning auch auf technischer Ebene zusammenzubringen. Der ontologiezentrierte Architekturentwurf strukturiert die unterschiedlichen Dienste.

Lernunterstützende Dienste

Für die Grundoperationen für das situationsgesteuerte Lernen wurden Dienste umgesetzt: die Kompetenzlückenberechnung, die kompetenzbasierte Lerngelegenheitsauswahl und die Zusammenstellung von Lernprogrammen. Im Gegensatz zu existierenden Ansätzen werden auch komplexere Kompetenzmodelle ausgenutzt, die Relationen wie Generalisierung zwischen Kompetenztypen und Komposition zwischen Kompetenzen unterstützen. Dadurch lässt sich auch Nutzung von Kompetenzen auf unterschiedlichen Abstraktionsniveaus und Granularitätsstufen unterstützen, indem eine Kompetenzsubsumtion in die

Berechnung mit einfließt. Diese Dienste sind dabei nicht auf das situationsgesteuerte Lernen beschränkt, sondern stellen Elementarbausteine für eine dienstorientierte Architektur für kompetenzorientierte Personalentwicklung dar.

Infrastruktur für Kontextgewinnung und -verwaltung

Mit der erarbeiteten Kontextinfrastruktur wurde die Grundlage geschaffen, um das Konzept der Situation auch tatsächlich zu operationalisieren. Sie trägt der Herausforderung Rechnung, dass die Gewinnung von Kontextinformationen i.a. unsichere Ergebnisse liefert, und sich Kontextinformationen über die Zeit hinweg ändern. Der benutzte Formalismus sucht den Kompromiss zwischen ausdrucksächtigen ontologiebasierten Formalismen, die allerdings schlecht mit temporalen und unsicheren Daten zurechtkommen, und einfachen relationalen Ansätzen, für die effiziente Indexstrukturen existieren. Durch das Konzept der Alterung wird eine Balance zwischen Vollständigkeit und Korrektheit/Zuverlässigkeit erreicht, was auch in anderen Bereichen relevant ist, etwa im Bereich der Ambient-Technologien (z.B. AAL).

Inhaltsverzeichnis

Definitionen	1
1. Einleitung	3
1.1. Motivation	3
1.2. Herausforderungen und Handlungsbedarf	5
1.3. Ziele	6
1.4. Überblick	7
I. Der Wissensreifungsprozess: Ein Prozessmodell für das arbeitsbegleitende Lernen	11
2. Arbeitsbegleitendes Lernen	15
2.1. Begriff des arbeitsbegleitenden Lernens	15
2.2. Begriff des E-Learning	17
2.3. Thesen zum arbeitsbegleitenden Lernen	18
2.4. Arbeitsbegleitendes Lernen hat viele Formen	19
2.5. Arbeitsbegleitende Lernprozesse sind nicht isoliert betrachtbar	20
2.6. Arbeitsbegleitendes Lernen braucht pädagogische Führung	23
2.7. Arbeitsbegleitendes Lernen ist stark von der Situation beeinflusst	24
2.7.1. Arbeitsbegleitendes Lernen und Personalentwicklung	26
2.7.2. Arbeitsbegleitendes Lernen und Wissensmanagement	26
2.7.3. Arbeitsbegleitendes Lernen und Geschäftsprozessmanagement	27
2.8. Modelle	28
2.8.1. Konstruktivismus als Lerntheorie	28
2.8.2. Modell für die Beziehung zwischen individuellen Lernprozessen und den Artefakten	32
3. Der Wissensreifungsprozess als Makromodell	35
3.1. Einführung	35
3.2. Exkurs: Wissensbegriff	36
3.2.1. Wissensbegriff dieser Arbeit	36

3.2.2.	Organisationales und Systemwissen	39
3.3.	Existierende Modelle	40
3.3.1.	SECI-Modell	40
3.3.2.	Snowdens Cynefin-Modell	42
3.3.3.	Lebenszyklus des Wissens	45
3.3.4.	Modell für organisationale Informationsverarbeitung nach Maier	45
3.4.	Wissensreifungsprozess: Grundidee und Phasen	46
3.5.	Reifungsschritte und die Rolle der Situation	48
3.6.	Brüche innerhalb des Prozesses und Ansätze zu deren Überwindung	50
3.7.	Fazit	53
II.	Situationsgesteuertes Lernen:	
	Eine Methode für die Unterstützung des Lernens bei Bedarf	55
4.	Unterstützung des Lernens bei Bedarf	59
4.1.	Lernen bei Bedarf und die Steuerung von Lernprozessen	59
4.1.1.	Begriff des Lernens bei Bedarf	59
4.1.2.	Steuerung von Lernprozessen	60
4.1.3.	Fokus der Arbeit	62
4.2.	Szenarien	62
4.2.1.	Szenario 1 – Kundenberater im Logistikvertrieb: Bedarfsorientierte Empfehlungen auf der Basis von Rollen	63
4.2.2.	Szenario 2 – Projektmanagement: Prozessorientiertes Lernen bei Bedarf durch Erkennen des Prozesskontextes	64
4.2.3.	Szenario 3 – Softwareentwicklung: Erkennen von aktuellen Aufgaben	65
4.3.	Existierende methodische Ansätze für das Lernen bei Bedarf	66
4.3.1.	E-Learning-Ansätze: pädagogisch motivierte Fremdsteuerung	66
4.3.2.	Geschäftsprozessorientiertes Wissensmanagement: organisational motivierte Fremdsteuerung	69
4.4.	Fazit	71
5.	Informationsrecherche als Lernprozess	73
5.1.	Grundbegriffe	73
5.1.1.	Suche und Recherche	73
5.1.2.	Informationsbedürfnis	74
5.1.3.	Akteure und ihre Aktionen	75
5.1.4.	Relevanz	77
5.2.	Modelle für das Informationsverhalten	78

5.2.1. Wilsons Modell des Informationsverhaltens	78
5.2.2. Modell des Informationsverhaltens nach Niedzwiedzka	80
5.2.3. Informationsverhaltensmodell nach Choo	82
5.3. Modelle für das Informationsrechercheverhalten	83
5.3.1. Kuhlthaus Informationsrechercheprozessmodell	83
5.3.2. Wang Baldonados Anwendung der Sensemaking-Theorie auf die In- formationssuche	86
5.4. Modelle für das Informationssuchverhalten	87
5.5. Just-in-Time Information Retrieval	88
5.6. Fazit	89
6. Methodik des situationsgesteuerten Lernens	93
6.1. Grundidee	93
6.2. Ablauf anhand der Beispielszenarien	95
6.2.1. Szenario 1: Kundenberater im Logistikvertrieb	96
6.2.2. Szenario 2: Projektmanagement	96
6.2.3. Szenario 3: Softwareentwicklung	97
6.3. Abstraktes Modell	97
6.3.1. Sicht des Lernenden	97
6.3.2. Sicht des Systems	98
6.4. Fazit	103
 III. Dienstinfrastruktur für das situationsgesteuerte Lernen	 105
7. Entwicklungsmethodik und Anforderungen für ein Dienstrahmenwerk	109
7.1. Funktionale Anforderungen	109
7.1.1. Funktionale Anforderungen aus der Beschreibung des Systemver- haltens	109
7.1.2. Funktionale Anforderungen: Kontextaspekte	110
7.1.3. Funktionale Anforderungen: Lerngelegenheiten	112
7.2. Nicht-funktionale Anforderungen	112
7.3. Methodische Vorgehensweise	113
7.4. Ontologiezentrierter, dienstorientierter Architekturstil	115
7.5. Fazit	117
8. Domänenontologie	119
8.1. Grundlegende Überlegungen	119
8.1.1. Kompetenzorientierter Grundansatz	120
8.2. Formalisierung des Kontextbegriffes und Kontextontologie	122

8.2.1.	Grundlegendes zum Kontextbegriff	122
8.2.2.	Existierende Ansätze zur Identifikation und Klassifikation von Kontextmerkmalen	124
8.2.3.	Entwicklung der Kontextontologie	130
8.3.	Lerngelegenheiten und Kompetenzen	139
8.3.1.	Lerngelegenheit	141
8.3.2.	Informationsontologie	142
8.3.3.	Kompetenzontologie	142
8.3.4.	Organisationsontologie	147
8.3.5.	Bezug zu anderen Modellen im Unternehmen	147
8.4.	Fazit	150
9.	Referenzarchitektur für situationsbewusste lernunterstützende Systeme	153
9.1.	Existierende Architekturen	153
9.1.1.	Kontextbewusste Anwendungen	154
9.1.2.	Adaptive E-Learning-Systeme	155
9.1.3.	Offene Architekturen für E-Learning-Systeme	156
9.1.4.	Enterprise Knowledge Infrastructures	157
9.2.	Grundstruktur der Referenzarchitektur	160
9.2.1.	Infrastruktur	160
9.2.2.	Situationsbewusste Dienste	161
9.3.	Dienste der Referenzarchitektur	164
9.3.1.	Infrastruktur und Datenhaltung	165
9.3.2.	Situationsbewusste Dienste	168
9.3.3.	Lernkoordination	170
9.3.4.	Adaption und Endbenutzerinteraktion	170
9.4.	Fazit: Verfeinerte Architektur	173
10.	Situationsbewusste Dienste für die Lernunterstützung	175
10.1.	Kompetenzlückenberechnung und Lerngelegenheitsselektion	176
10.1.1.	Problemanalyse	177
10.1.2.	Kompetenzrelationen und Kompetenzsubsumtion	180
10.1.3.	Analyse der Kompetenzlücke	186
10.1.4.	Kompetenzbasierte Lerngelegenheitsauswahl	189
10.2.	Lernprogramme zusammenstellen	194
10.2.1.	Stand der Technik: Semantische Komposition von Lernprogrammen	195
10.2.2.	Verfahren zur Zusammenstellung von Lernprogrammen	195
10.3.	Lernkoordination	198
10.4.	Fazit	198

11. Adaption von Endbenutzeranwendungen	201
11.1. Integration in Lernumgebungen	202
11.1.1. SCORM-kompatible Lernmanagementsysteme (LMS)	202
11.1.2. Ausführen von generierten Lernprogrammen	203
11.1.3. Kontextbewusste Lernobjekte: Bereitstellung von Kontextinformati- on für Lernobjekte	205
11.2. Lernassistent: Anzeige von Lernempfehlungen	208
11.3. Fazit	210
IV. Kontextinfrastruktur	213
12. Problemanalyse, Anforderungen und Lösungsansatz	217
12.1. Anforderungen	218
12.1.1. Funktionale Anforderungen	218
12.1.2. Nicht-funktionale Anforderungen	220
12.2. Stand der Technik: Kontextverwaltung	220
12.2.1. Architektur von Kontextinfrastrukturen	221
12.2.2. Datenmodelle für Kontextinformationen	224
12.3. Lösungsansatz und Architektur	228
12.3.1. Grundprobleme der Architektur	228
12.3.2. Überblick über die Architektur	229
12.4. Fazit	232
13. Kontextfaktenebene	233
13.1. Datenmodell für Kontextfakten	234
13.2. Alterung	236
13.2.1. Existierende Ansätze zur Alterung	236
13.2.2. Repräsentation und Semantik von Alterung	237
13.3. Anfragen an den Kontextfaktendienst	238
13.3.1. Grundstruktur und Semantik der Anfragen	239
13.3.2. Anfragen mit Ontologien auf Datentypenebene	241
13.4. Anfragebearbeitung auf der Basis relationaler Datenbanksysteme	242
13.4.1. Auswertung der Alterungsfunktionen	243
13.4.2. Auswertung der Ontologieoperatoren	244
13.5. Kontinuierliche Anfragen	245
13.6. Fazit	247
14. Kontextinformationsebene	249
14.1. Schema für Kontextinformationen	249

14.2. Aggregation von Fakten und Konfliktauflösung	250
14.2.1. Begriff des Konflikts	251
14.2.2. Konfliktauflösungsstrategien	252
14.3. Anfragen	254
14.3.1. Anfragen nach Kontextmerkmalsausprägungen (<i>ContextQuery</i>) . . .	255
14.3.2. Überprüfung der Gültigkeit von Fakten (<i>CheckQuery</i>)	255
14.3.3. Anfragen nach Benutzern (<i>UserQuery</i>)	257
14.4. Kontinuierliche Anfragen	258
14.5. Verteilung	260
14.6. Fazit	261
V. Evaluierung und Schluss	263
15. Implementierung und Evaluierung von Architektur und Diensten	267
15.1. Evaluierung des Wissensreifungsprozessmodells	267
15.1.1. Qualitative Untersuchung	268
15.1.2. Untersuchung anhand der Wikipedia	270
15.2. Implementierung	273
15.2.1. Überblick und Architektur des Prototypen	274
15.2.2. Endbenutzerinteraktion	274
15.2.3. Lernkoordination und Situationsbewusste Dienste	276
15.2.4. Infrastruktur und Datenhaltung	277
15.2.5. Integration und Externe Informationsquellen	278
15.2.6. Administrationswerkzeuge	279
15.3. Konfiguration der Demonstratoren	279
15.4. Evaluation	280
15.4.1. Vorüberlegungen	280
15.4.2. Phase 1 - Formative Evaluierung	281
15.4.3. Phase 2: Summative Evaluierung	282
15.5. Fazit	284
16. Zusammenfassung und Ausblick	293
16.1. Wissensreifungsprozess	293
16.2. Methodik des situationsgesteuerten Lernens	295
16.3. Referenzontologie und Referenzarchitektur für die arbeitsbegleitende Lern- unterstützung	295
16.4. Lernunterstützende Dienste	296
16.5. Kontextbewusste Lernobjekte	297

16.6. Kontextgewinnung und -verwaltung	297
Abbildungsverzeichnis	299
A. Basisontologie in OWL	305
B. Schema für die relationale Datenbank des Kontextfaktendienstes	311
C. UML-Diagramme Kontextverwaltungsdienst	315
D. Beispiel-Protokoll formative Evaluierung	319
Literaturverzeichnis	320

Danksagung

Ob das alte Sprichwort »Was lange währt, wird endlich gut« auch hier gilt – das muss der Leser entscheiden. In jedem Fall spiegelt die vorliegende Arbeit den größten Teil der Ergebnisse meiner wissenschaftlichen Tätigkeiten am FZI Forschungszentrum Informatik wider, wo ich zunächst als wissenschaftlicher Mitarbeiter und dann als Abteilungsleiter für die Gruppe von Professor Lockemann tätig war, der immer bemüht war, die Arbeit trotz der schwierigen zeitlichen Dreifachbelastung aus Abteilungsleitung, Projektakquise und -arbeit und in europäischen Forschungsprojekten wie Industrieprojekten und Fertigstellung Dissertation immer wieder vorangetrieben hat und in intensiver Beschäftigung mit ihr die Qualität und Stringenz maßgeblich gefördert hat. Daneben ist auch den Kollegen zu danken, die erst das kreative Umfeld geschaffen haben, innerhalb dessen eine Arbeit wie diese entstehen konnte. Besonders hervorzuheben sind hier Dr. Wassili Kazakos, Dr. Gabor Nagypal, Heiko Paoli, Peter Tomczyk, im besonderen aber auch Simone Braun, mit der zusammen im Rahmen der Projekte Im Wissensnetz und MATURE etliche Aspekte weiter vertieft werden konnten, was sich in zahlreichen gemeinsamen Publikationen niederschlug.

Der wesentliche Kern der Arbeit wurde im EU-Projekt LIP (Learning in Process) gelegt. Hier möchte ich besonders Gabriel Fractman Lazaro von META4 für die Zusammenarbeit in der Projektkonzeption und bei manch schwierigen Projektsituationen, aber auch John Cook und Claire Bradley vom LTRI der London Metropolitan University danken, die die formative und summative Evaluation des Systems koordiniert haben.

Auch wenn die letztendlich erfolgreiche Beantragung des Integrierenden EU-Projektes MATURE nicht zur Beschleunigung der Fertigstellung der Arbeit beigetragen hat, so waren die Diskussionen rund um die Wissensreifungsthematik unglaublich motivierend, und ich danke allen Projektbeteiligten für die sehr engagierten gemeinsamen Aktivitäten, die den Wert von interdisziplinärer Zusammenarbeit anschaulich aufgezeigt haben. Hier ist neben Prof. Knut Hinkelmann (FHNW), Dr. Tobias Ley (Know-Center), Dr. Stefanie Lindstaedt (Know-Center), Prof. Ronald Maier (Universität Innsbruck) und Dr. Uwe Riss (SAP) insbesondere Prof. Johannes Magenheim von der Universität Paderborn hervorzuheben, der dankenswerter Weise das Korreferat übernommen hat und eine stärker sozialwissenschaftliche Perspektive eingebracht hat.

Die Übertragbarkeit von etlichen Aspekten der Arbeit (wie z.B. der ontologiezentrierte Entwicklungsansatz und das Kontextmanagement) ließ sich im Rahmen des Integrierten

EU-Projekt SOPRANO, das sich mit Ambient Assisted Living beschäftigt, hervorragend zeigen. Hier gilt der Dank für die vielfältigen fruchtbaren Diskussionen vor allem meinem Kollegen Peter Wolf, Dr. Michael Klein und Dr. Dirk Balfanz von der CAS AG sowie Anastacia Garbi und Elena Avatangelou von EXODUS für die motivierende Atmosphäre und die Freiräume in der Umgestaltung des Projektplanes.

Letztendlich ist auch den Abteilungsleiterkollegen Dr. Andreas Abecker und Dr. Clemens van Dinther zu danken, die den in den letzten drei Jahren sehr schwierigen Spagat zwischen Dissertation und Projekt-/Akquisegeschäft ermöglicht haben, und vor allem auch Carsten Holtmann, der mich als Bereichsleiter von IPE von zeit- und energieraubenden Verwaltungsakten merklich entlastet hat.

Der sicherlich wichtigste Dank gebührt letztendlich meiner Frau Christine, die mir die zeitlichen Freiräume zusätzlich zur intensiven Reisetätigkeit gewährt und geschaffen hat und wesentlich auch die nicht-technischen Aspekte der Arbeit (besonders im Bereich des Kompetenzmanagement sowie der organisatorischen Einbettung insgesamt) zu einer stärker ganzheitlichen Perspektive gefördert hat.

Definitionen

- Alterungsannotation, **238**
- Alterungsfunktion, **238**
- Anfrage an Kontextfaktendienst und deren Semantik, **240**
- Arbeitsbegleitendes Lernen, **16**

- Benutzungskontext, 123
- Benutzungssituation, **94**

- Datentyp, **234**

- E-Learning, **17**
- Elektronische Portfolios, 21

- Hintergrundwissen, 40

- Informationsbedürfnis, 74
- Informationsontologie, 142
- Informationsrechercheverhalten, 73
- Informationssuchverhalten, 74
- Informationsverhalten, 73

- Kompetenz, **144**
- Kompetenzkatalog (formal), **183**
- Kompetenzmanagement, 26
- Kompetenzskalen, 145
- Kompetenzsubsumtion, **180**
- Konfliktauflösung, **252**
- konfliktfrei, **251**
- Kontext, **123**
- kontextbewusst, 124
- kontextbewussten Lernobjekten, 205

- Kontextfaktenschema, **235**
- Kontextinformationsschema, **250**
- Kontextmerkmal, **123, 234**
- Kontextmerkmalswert, **250**

- Lerngelegenheiten, 62
- Lernobjekte, 142
- Lernunterstützung, 95

- Merkmalsstruktur, **234**

- Ordnung auf Lernobjekten, **197**
- organisationalen Wissens, 39

- pädagogische Agenten, 67

- semantisch kohärent, 113
- Situation, **94**
- situationsbewusst, 124
- Skills-Management, 26
- subjektive Relevanz, 77

- Verträglichkeit der Merkmalsstruktur, **235**

- Wissen, **39**

1.

Einleitung

1.1. Motivation

Wir sind auf dem Wege in eine Lerngesellschaft. Die Schlagworte der letzten Jahre wie »lebenslanges Lernen«, »lernende Organisation« oder auch »Wissensgesellschaft«, »Wissensmanagement« oder »Wissensarbeiter« zeigen, wie wichtig das Aneignen neuer Kenntnisse, Fähigkeiten und Fertigkeiten geworden ist. Das streng organisierte und institutionalisierte Lernen auf Vorrat in Schule und Ausbildung in den ersten Lebensphasen dringt zunehmend gegenüber dem kontinuierlichen Lernen im Berufs- wie im Privatleben in den Hintergrund. Weil die Anforderungen an die individuelle Handlungskompetenz immer komplexer und weniger vorhersagbar werden, ist ein Lernen bei Bedarf erforderlich (im übrigen in Analogie zu parallelen Entwicklungen in Richtung »just-in-time«-Produktion). Dieses kann aber nur dann stattfinden, wenn Lernen in die alltäglichen Aktivitäten eingebettet wird und nicht mehr als komplett separater Prozess aufgefasst wird. Im beruflichen Kontext bedeutet dies insbesondere, dass Lernen und Arbeiten als miteinander verwobene Aktivitäten verstanden werden müssen (vgl. [Str05]).

Durch diese Einbettung in andere Prozesse wird Lernen allerdings viel individueller. Während in Seminaren die Bildungsprozesse der einzelnen Teilnehmer noch einigermaßen synchron ablaufen, kann dies bei integriertem Lernen nicht mehr vorausgesetzt werden. Dies steigert allerdings die Komplexität solcher Bildungsprozesse. Eine Lösung ist es sicherlich, Lernen grundsätzlich in den Verantwortungsbereich des Einzelnen zu legen. Dies bedeutet, komplett auf das Prinzip des Führens zu verzichten. In der betrieblichen Bildung würde man damit zum einen die pädagogische Führung aufgeben, die auf der Erkenntnis beruht, dass für das Lernen ein gewisses Maß an Geführtwerden eine große Bedeutung hat, zum anderen würde man aber auch der an den Unternehmenszielen ausgerichteten Personalentwicklung die Grundlage entziehen. Der komplette Rückzug auf ein selbstgesteuertes und ungeführtes Lernen erscheint deshalb mittel- und langfristig weder aus indi-

vidueller noch organisationaler Sicht sinnvoll und wünschenswert. Allerdings muss man dann die Komplexität der individualisierten Bildungsprozesse beherrschbar machen.

Hierfür bieten sich Informationstechnologien an, die inzwischen zu einem integralen Bestandteil der Arbeitswelt geworden sind. Und dementsprechend werden sie auch bereits im großen Stile dazu eingesetzt, das organisierte Lernen zu verwalten (Lernmanagementsysteme), die Kompetenzen der Mitarbeiter bewusst zu gestalten (Skills-Management-Systeme) oder den Zugang zu im Unternehmen vorhandenen Ressourcen zu verbessern (Informations- und Wissensmanagementsysteme). Allerdings ist es bislang noch nicht zufriedenstellend gelungen, Lernen und Arbeiten wirklich zu verbinden. Bei lernorientierten Systemen liegt der Fokus immer noch zu sehr auf dem »Kursparadigma«, das sich als Virtualisierung von Präsenzseminaren versteht; bei informationsorientierten Systemen herrscht eine noch zu große Naivität bezüglich der Vorstellung vor, wie sich ein Mitarbeiter Wissen aneignet (und wie nicht).

Das Kernproblem ist hierbei, dass all diese Systeme zuwenig die Situation berücksichtigen, in der sie benutzt werden. Wenn Lernen in (und nicht neben) den Arbeitsprozessen stattfinden soll, dann kann das Systemverhalten nicht ignorieren, um was es in dem Arbeitsprozess gerade geht, wie die Beziehungen zum organisationalen Umfeld aussehen und wie groß der objektive Zeitdruck ist. Und wenn es wirklich um Lernen als einen individuellen Wissensaufbau geht, dann kann das Systemverhalten ebenso nicht ignorieren, dass Vorwissen, persönliche Ziele, Lernpräferenzen, emotionale Verfassung u.v.m. das Lernen elementar beeinflussen. Was also benötigt wird, sind situationsbewusste lernunterstützende Dienste.

Diese Forderung nach Situationsbewusstsein von IT-Diensten für die Lernunterstützung fällt zusammen mit technischen Entwicklungen im Bereich der Informationssysteme. Nachdem durch die Konvergenz der Kommunikationsmedien und die zahlreichen Bemühungen zur Integration unterschiedlicher Informationssysteme unter einheitlichen Schnittstellen wie z.B. dem World-Wide Web immer mehr und immer größere Informationsbestände über das Netz verfügbar wurden, tritt die Frage nach der Verfügbarkeit zunehmend in den Hintergrund. Stattdessen wird es für einen Benutzer immer wichtiger, das von ihm Gesuchte hinreichend genau zu spezifizieren, um eine überschaubare Ergebnismenge zu erhalten. Doch die ihm abverlangte Präzision steht in den meisten Fällen in einem direkten Widerspruch zur Forderung nach Einfachheit der Anfrageformulierung. Dies führt in der Praxis dazu, daß auf bestimmte Aufgaben spezialisierte Anfrageschnittstellen benutzt werden, die die Flexibilität der Systeme stark einschränken. Erschien in der Vergangenheit die Verbesserung der technischen Effizienz (Speicherbedarf, Performanz in der Anfragebearbeitung) die wichtigste Aufgabe, ist heute eher die Verbesserung der Nutzungseffizienz aus Benutzersicht vorrangig. Denn der Mensch mit seiner beschränkten Aufnahmefähigkeit ist zum beschränkenden Element geworden, nicht mehr das technische System, was mit dem Schlagwort der »Informationsüberflutung« plakativ zum

Ausdruck gebracht werden soll.

Statt zwischen den Einschränkungen spezialisierter Schnittstellen oder aber dem hohen Einarbeitungsaufwand und Nutzungsaufwand allgemeiner Schnittstellen wählen zu müssen, wird unter dem Stichwort »context-awareness« der Ansatz verfolgt, Informationen über den Benutzer und seine aktuelle Situation miteinzubeziehen. Dies ist motiviert durch die Beobachtung der zwischenmenschlichen Kommunikation, bei der auch der Kommunikationskontext den eigentlichen Austausch der Nachrichten entlastet. Das Informationsbedürfnis eines Benutzers ist ja immer eingebettet in das, was er gerade tut und wohin er gerade will und was er bereits weiß.

1.2. Herausforderungen und Handlungsbedarf

So bestechend die Idee von einer situationsbewussten Lernunterstützung ist, so wirft sie bei näherem Hinsehen grundsätzliche Fragen auf, die sowohl auf der Anwendungsebene, als auch auf der technischen Ebene ihren Ursprung haben. Präziser gesagt: diese Problemereiche hängen eng miteinander zusammen, da das Verständnis für die sinnvolle Anwendung von Situationsbewußtsein für die Lernunterstützung die Anforderungen an die technische Infrastruktur definiert und umgekehrt die technische Infrastruktur durch ihre Möglichkeiten umreißt, was sich realisieren läßt und dementsprechend methodisch sinnvoll einzusetzen ist. Deshalb sollen im Rahmen dieser Arbeit beide »Welten« zusammen betrachtet und auf beiden Ebenen Lösungen entwickelt werden, auch wenn der Blickwinkel doch weiterhin eher technisch geprägt sein wird.

Betrachtet man den Bereich des arbeitsbegleitenden Lernens näher, so zeigt sich, dass trotz der Forschungsansätze im Bereich der adaptiven Lernsysteme und der Intelligenten Tutoriellen Systeme – die oft zu technologiegetrieben waren – folgende fundamentalen Fragen noch immer ungeklärt sind:

- Welche Rolle hat die Situation des Lernenden für das arbeitsbegleitende Lernen? Welche Elemente der Situation beeinflussen welche Aspekte des Lernens?
- Wie ist die Beziehung zwischen individuellem Lernen und der Nutzung von externen Ressourcen (einschließlich Personen im sozialen Umfeld)?
- Wie sollen Methoden für eine situationsbewusste Lernunterstützung aussehen? Welche Beziehung haben sie zur Suche nach Informationen?

Hier gilt es, einen allgemeinen Rahmen zu schaffen, der die Beantwortung dieser Fragen in Beziehung setzen läßt zur technischen Realisierung in Form von konkreten Diensten. Dieser Rahmen liefert gleichzeitig die Anforderungen an die technische Infrastruktur.

Bei der technischen Umsetzung ist das kritische Element eine geeignete Infrastruktur, die die Situation des Benutzers systemseitig verfügbar macht. Diese technische Infrastruktur sieht sich aber – trotz der langjährigen Beschäftigung mit der Thematik – zwei fundamentalen Herausforderungen gegenüber:

- Ein Kernproblem, dem bislang in den meisten Ansätzen zu wenig Beachtung geschenkt wurde, stellt dar, dass die Situation dem System nicht unmittelbar zugänglich ist, sondern es sich mit einem **unvollkommenen Abbild** (was im weiteren Verlauf der Arbeit als »Kontext« bezeichnet wird) zufriedengeben muss, das aus einer Kombination von unterschiedlichen indirekten Verfahren aufgebaut werden muss (was an sich schon ein schwieriges Unterfangen ist). Hieraus ergeben sich zwei Konsequenzen: zum einen muss beim Entwurf von Systemen klar sein, dass man eben nur ein unvollkommenes Abbild zur Verfügung hat, zum anderen erfordert der Umgang mit den einzelnen Bestandteilen des Abbildes ebenfalls angemessene Verfahren, die mit Unvollkommenheiten umgehen können.
- Ein verschärfendes Problem liegt in der **Zeitabhängigkeit** des Situationsbegriffs begründet: Situationen ändern sich, so dass einmal gesammelte Informationen nicht ohne weiteres später wiederverwendet werden können. Umgekehrt ist in den meisten Fällen ein solches Sammeln unumgänglich. Eine nähere Analyse des Problems zeigt, dass Zeitabhängigkeit für unterschiedliche Teile der Situation höchst differenziert betrachtet werden muss. Ändern sich persönliche Daten wie Namen oder Geburtsdatum überhaupt nicht, so ändern sich Interessen zumindest mittelfristig, Aufenthaltsort, emotionaler Zustand oder aktuelle Aufgabe aber sogar relativ schnell. Bislang haben sich hier technische Infrastrukturen darauf zurückgezogen, dass sie entweder recht stabile Benutzerprofile nutzen oder die Ermittelbarkeit bei Bedarf postuliert haben (analog zum Sensormodell).

Daraus ergibt sich im wesentlichen ein *Verwaltungsproblem*, das nicht allein auf das Szenario des arbeitsbegleitenden Lernens beschränkt ist, sondern vorgefunden werden kann, wo immer mit Informationen über die Situation des Benutzers umgegangen wird und sich diese nicht auf Ort oder sensornahe Größen beschränkt. Dieses Verwaltungsproblem ist der Schlüssel für die Anwendung situationsbewusster Techniken im großen Stil, wird aber in den derzeitigen Forschungsbemühungen weitgehend ignoriert.

1.3. Ziele

In dieser Arbeit geht es darum, die Forderungen nach mehr Situationsbewusstsein aus der Problematik des arbeitsbegleitenden Lernen heraus zusammenzubringen mit der tech-

nischen Umsetzung von situationsbewussten Informationsdiensten. Ziel ist die Verbesserung des arbeitsbegleitenden Lernen durch Bereitstellung von Diensten, die unter Berücksichtigung der aktuellen Situation des Nutzers bzw. Lernenden die Lernprozesse unterstützen. Dieses Gesamtziel gliedert sich in zwei wesentliche Teilziele:

- eine Untersuchung, wie durch Situationsbewußtsein von Diensten arbeitsbegleitendes Lernen unterstützt werden kann, welche Dienste im Grenzbereich der heutigen Wissensmanagement- und E-Learning-Ansätze hierfür benötigt werden und wie sie in eine Rahmenarchitektur einzuordnen sind, sowie die Realisierung exemplarischer Dienste
- und die Konzeption einer Infrastruktur zum Umgang mit Kontextinformationen als systemseitigen Abbild der Situation von Benutzern, und zwar basierend auf den Anforderungen, die sich aus den lernunterstützenden Diensten ergeben.

Als Hintergrund und Fallstudie dient das von der EU geförderte Verbundprojekt »Learning in Process« und darauf aufbauende Folgeprojekte ([SW03], [SW04], [Sch04a], [Sch04b], [Sch05a], [SB06]), in deren Rahmen bereits Kernelemente dieser Arbeit veröffentlicht wurden.

1.4. Überblick

Die der Arbeit zugrundeliegende Forschungsmethodik ist *Design Research*, deren wesentliche Ergebnisse (vgl. [MS95], [WWS92]) ein konzeptuelles Vokabular (*constructs*), Modelle (*Aussagen über Beziehungen zwischen den Konzepten*), Methoden und deren Umsetzung (*Instantiations*) sind. Die Vorgehensweise zur Erzielung dieser Ergebnisse orientiert sich an Modellen wie [TVTY90], nach denen sich der Erkenntnisprozess in fünf Schritten vollzieht: Entwickeln eines Problembewusstseins (*awareness of problem*), Lösungsvorschlag (*Suggestion*), Lösungsentwicklung und Implementierung (*Development*), Evaluation und Schlussfolgerung (*Conclusion*).

Da der Hauptschwerpunkt der Arbeit auf der Lösungsentwicklung liegt, gliedert sich die Arbeit in fünf Hauptteile (vgl. Abb. 1.1):

- Der erste Teil widmet sich als Beitrag zur Schaffung des Problemsbewusstseins dem arbeitsbegleitenden Lernen und dem notwendigen Verständnis für die Zusammenhänge und Probleme, um so die Grundlage zu schaffen für die Konzeption von lernunterstützenden Diensten. Dabei geht in einem ersten Schritt um ein geeignetes Makromodell, das es erlaubt, das Themenfeld zu strukturieren, Beziehungen zwischen den individuellen Lernprozessen und den überindividuellen Abläufe zu schaffen

und die Rolle der Situation im Großen zu verankern. Hierzu wird der **Wissensreifungsprozess** entwickelt. Dieses Modell zeigt besondere Problembereiche, die einem effizienten Umgang mit Wissen und Kompetenzen behindert. Darauf aufbauend kann auch die Bedeutung von Informationen innerhalb dieses Prozesses und die Beziehung zur Informationssuche herausgearbeitet werden.

- Im zweiten Teil als Kernlösungsvorschlag wird für einen der Problembereiche im Wissensreifungsprozess ein methodischer Rahmen für die Lernunterstützung entwickelt: das **situationsgesteuerte Lernen**.
- Dieser Rahmen bildet die Grundlage für ein technisches Rahmenwerk für die Lernunterstützung im dritten Teil; er liefert die Anforderungen für die Konzeption von lernunterstützenden situationsbewussten Diensten und die Entwicklung einer **dienstorientierten Architektur**, die eine situationsbewusste Lernunterstützung ermöglicht. Dabei geht es um die Identifikation von benötigten Diensten und die Spezifikation von Dienstschnittstellen und Interaktionsmustern. Auch das benötigte konzeptionelle Modell wird entwickelt, das die semantische Kohärenz der Dienste sicherstellt. Die technische Realisierung dieser situationsbewussten Dienste liefert die Anforderungen an die Kontextinfrastruktur.
- Als zweiter Baustein der Lösungsentwicklung kümmert sich der vierte Teil um eine **Kontextverwaltungsinfrastruktur**, die die Basis für die lernunterstützenden Dienste darstellt. Dabei werden als zentrale Herausforderungen die Unvollkommenheit und die Dynamik von Kontextinformationen identifiziert. Für die Bewältigung dieser Herausforderungen werden ein Kontextmodell als Datenmodell sowie darauf aufbauende Implementierungstechniken vorgestellt, die effizient eine qualitativ hochwertige Sicht auf den aktuellen Kontext des Benutzers liefern.
- Im fünften Teil erfolgt schließlich die Implementierung von Demonstratoren und die Evaluierung ihres Einsatzes in realen Unternehmensumgebungen.



Abbildung 1.1.: Gesamtüberblick über die Arbeit

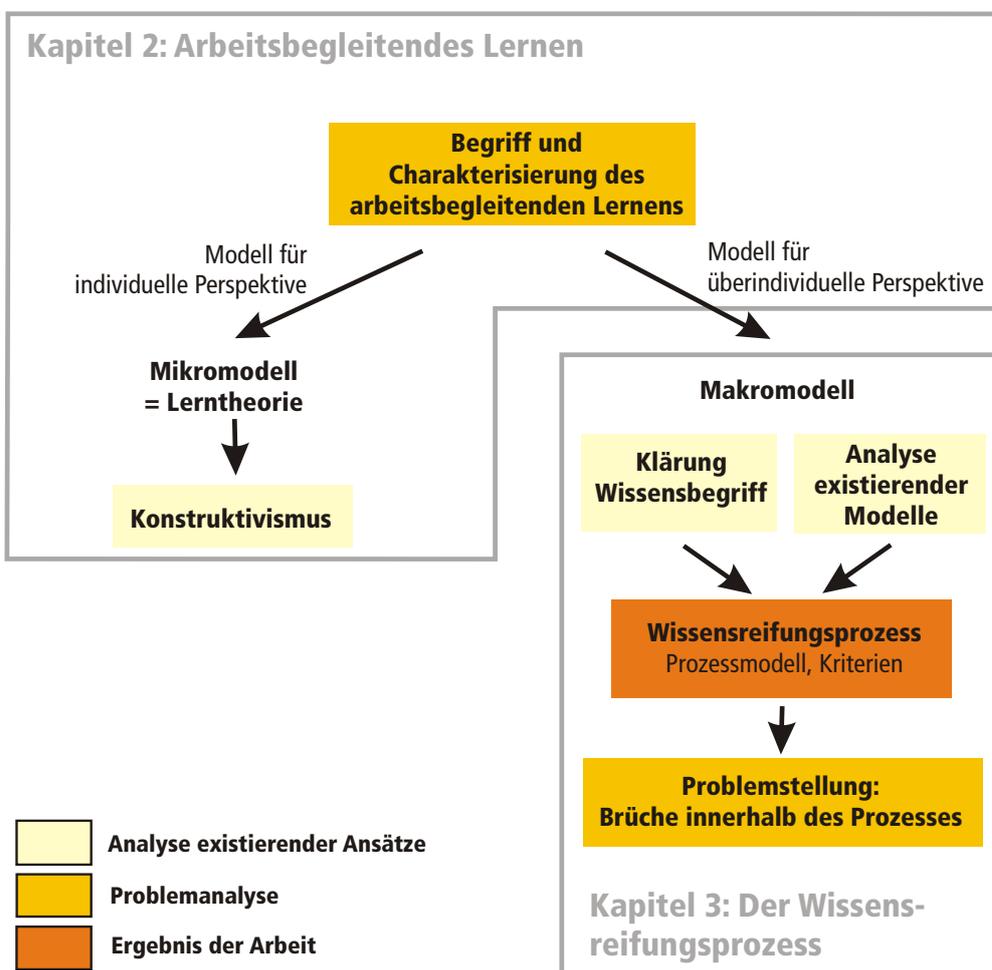
Teil I.

Der Wissensreifungsprozess: Ein Prozessmodell für das arbeitsbegleitende Lernen

Problemstellung

Um eine methodische und/oder informationstechnische Unterstützung für das arbeitsbegleitende Lernen konzipieren und umsetzen zu können, wird ein Modell benötigt, das die wesentlichen Eigenschaften herausarbeitet und Ansatzpunkte identifiziert. Dieses Modell muss Rahmenbedingungen und Anforderungen für die Unterstützungsmethodik und deren Umsetzung liefern.

Vorgehensweise und Struktur



2.

Arbeitsbegleitendes Lernen

In diesem Kapitel geht es darum, den Begriff des arbeitsbegleitendes Lernens zu präzisieren und die wesentlichen Eigenschaften herauszuarbeiten, die die Grundlage für die weitere Arbeit bilden. Dies erfolgt auf der Basis von griffigen Thesen, die anhand der wissenschaftlichen Literatur belegt werden. Daran anschließend werden geeignete Modelle identifiziert bzw. der Handlungsbedarf für die Erarbeitung solcher Modelle dargelegt, was zum nächsten Kapitel überleitet und dieses motiviert.

2.1. Begriff des arbeitsbegleitenden Lernens

Obwohl arbeitsbegleitendes Lernen an sich nichts Neues darstellt, sondern in der beruflichen Ausbildung seit jeher eine wichtige Rolle gespielt hat (z.B. Lehrlingsausbildung), ist in der letzten Zeit das Interesse sprunghaft angestiegen. Dies hängt damit zusammen, dass sich das traditionelle Weiterbildungssystem grundlegend in Richtung einer flexiblen Kompetenzentwicklung bewegt [SR02]. So stehen immer weniger formalisierte Lernkontexte im Vordergrund, und es wird bewusst nach einer umfassenden Integration von Lernen und Arbeiten gesucht. Auch wird der Mitarbeiter selbst zum Gestalter seiner Lernprozesse, die immer spezifischer werden.

Der Begriff des »arbeitsbegleitenden Lernens« wird in der wissenschaftlichen Literatur noch relativ selten verwendet. Üblicherweise wird das Lernen in Unternehmenskontexten unterschieden nach der Beziehung des Lernortes zum Arbeitsplatz [Kir04]:

- **arbeitsgebundenes oder arbeitsimmanentes Lernen** (Lernort und Arbeitsplatz sind identisch)
- **arbeitsverbundenes Lernen** (räumliche und organisatorische Verbindung zwischen Lernort und Arbeitsplatz)

- **arbeitsorientiertes oder arbeitsbezogenes Lernen** (räumliche und organisatorische Trennung, nur noch didaktische Verbindung)

Diese Fixierung auf den rein räumlich-organisatorischen Aspekt macht unter der Zielsetzung einer besseren Integration von Lernen und Arbeit nur beschränkt Sinn, obwohl das z.B. auch in [BSS01] als Klassifikationskriterium herangezogen wird. Die räumlich-organisatorische Trennung ist meist nur Ausdruck einer fehlenden Unmittelbarkeit des Lernens in Bezug auf die Arbeitsprozesse. Stattdessen wird im folgenden der Begriff des arbeitsbegleitenden Lernens verwendet, der neben räumlichen und organisatorischen Kriterien auch das Inhaltliche berücksichtigt und auch für [Kir04] eine passende Bezeichnung darstellt:

Definition 2.1 (Arbeitsbegleitendes Lernen) *Arbeitsbegleitendes Lernen bezeichnet eine enge Verknüpfung von Lernen und Arbeiten, bei denen die Lernprozesse meist in räumlicher Nähe zum Arbeitsplatz und in zeitlicher Nähe zu den konkreten Arbeitsprozessen stattfinden, zu denen ein inhaltlicher Bezug besteht.*

Diese Definition lehnt sich an [SR02] an, wo aufbauend auf eine Definition von [DRM98] »arbeitsbegleitendes Lernen« als Lernen definiert wird, das »in relativer Nähe zum Arbeitsplatz und konkreten Arbeitsgeschehen mit inhaltlich größtmöglichem Praxisbezug realisiert wird«. Im Gegensatz zur dortigen Definition wurde allerdings etwas klarer herausgearbeitet, was unter »relativer Nähe [...] zum konkreten Arbeitsgeschehen« zu verstehen ist. STEGMAIER [Ste00] spricht hier auch von »arbeitsintegriertem Lernen«¹ und hebt die Situietheit in authentischen Problem- und Handlungssituationen als entscheidendes Element hervor. »Arbeitsbegleitendes Lernen« dürfte im wesentlichen auch das bezeichnen, was IBM in seiner E-Learning-Strategie neuerdings als »eingebettetes Lernen« (*embedded learning*) bezeichnet [Str05].

Im angelsächsischen Sprachraum ist die Terminologie — wie beim Themenfeld berufliche Bildung generell üblich — anders gelagert, aber ähnlich unübersichtlich. Die eheste Entsprechung findet sich im Begriff »**workplace learning**«, der von [Rot02] definiert wird als die Art des Lernens, die bei der Vorbereitung, Durchführung und Nachbereitung der Arbeit stattfindet. Hier ist der Arbeitsbezug inhaltlich gegeben, wobei implizit auch eine zeitliche Nähe enthalten ist, so dass »workplace learning« als englische Entsprechung angesehen werden kann. Allerdings zeigt sich in der derzeitigen Diskussion, dass zur Betonung der Integration von Lern- und Arbeitsprozessen der Begriff »work-integrated learning«, wie er z.B. von [SSÖ03] oder [USLG06] in Abgrenzung zu rein arbeitsprozessorientierten, aber nicht arbeitsprozessintegrierten Lernformen benutzt wird, eine präzisere Entsprechung ist.

¹Dem Begriff »arbeitsbegleitendes Lernen« wurde dennoch der Vorzug gegeben, weil »arbeitsintegriert« suggeriert, dass nicht die Möglichkeit besteht, den Arbeitsprozess zu verlassen.

2.2. Begriff des E-Learning

Arbeitsbegleitendes Lernen umfasst sowohl traditionelle Lernformen wie Unterweisung am Arbeitsplatz u.ä. als auch »neue Lernformen«, die üblicherweise unter dem Schlagwort »E-Learning« subsumiert werden. Da für diese Arbeit stärker E-Learning-Ansätze im Vordergrund stehen (aufgrund der Konzentration auf Lernunterstützung durch Informationsdienste), soll im folgenden kurz geklärt werden, was darunter zu verstehen ist und inwieweit eine Beschränkung auf E-Learning sinnvoll ist.

Obwohl die Unterstützung von Lernprozessen durch rechnergestützte Programme und Systeme eine lange Tradition hat, kam der Begriff des »E-Learning« erst im Gefolge der Internet-Euphorie Ende der 1990er Jahre auf. Anders als Begriffe wie »Computerunterstütztes Lernen« hat »E-Learning« keine wissenschaftliche Fundierung und keine hinreichend präzise Definition, sondern beschreibt wie die anderen marketinglastigen »E-Terms« (E-Commerce, E-Business, E-Human Resources, E-Health, etc.) einen Trend zur Umgestaltung traditioneller Prozesse durch konsequente Nutzung von Informations- und Kommunikationstechnologien [BBSS01].

In den letzten Jahren wurde versucht, den inzwischen allgegenwärtig gewordenen Begriff gegenüber anderen Begriffen abzugrenzen. Bei der Analyse lassen sich mehrere Ansätze erkennen: Zum einen wird davon ausgegangen, dass durch Internet-Technologien eine neue Qualität des Lernens mit elektronischen Medien entstanden ist und somit der Begriff E-Learning sich durch zusätzliche Eigenschaften vom rechnerunterstützten Lernen abhebt (z.B. [Wil03], [MHP05]). Etwas nüchterner wird z.B. im St. Galler Referenzmodell E-Learning eher als eine neue Bezeichnung für das computergestützte Lernen aufgefasst [BBSS01]. Ein weiterer Unterschied besteht in der Frage, ob E-Learning nur solche Lernformen bezeichnet, die ausschließlich am Rechner oder sogar über Netzwerke (»Online Learning«) stattfinden, oder ob jede Form von Lernprozessen, die zu ihrer Unterstützung Informationstechnologien einsetzen, zu E-Learning gehören.

Für diese Arbeit erscheint folgende Definition am sinnvollsten:

Definition 2.2 (E-Learning) *E-Learning wird in dieser Arbeit als rechnerunterstütztes Lehren und Lernen verstanden. Die Unterstützung kann dabei dadurch geschehen, dass elektronisch vorliegende Ressourcen (Lernprogramme o.ä.) genutzt werden oder rechnerunterstützte Kommunikations- und Zusammenarbeitsmöglichkeiten zur Verfügung gestellt werden, die Lernprozesse stimulieren und fördern.*

Hierbei ist es wichtig festzuhalten, dass diese Definition nicht nur Lernprozesse umfasst, die ausschließlich am Rechner stattfinden, sondern auch die bloße Unterstützung von Lernprozessen durch die Nutzung von Informationstechnologien. Diese Sichtweise teilen auch SEUFERT UND EULER [SE05]: »eLearning kann somit definiert werden als die

Anwendung von Informations- und Kommunikationstechnologien im Lernprozess.« Hierbei unterscheiden die Autoren die in obiger Definition unterschiedenen Anwendungsfälle als »Unterstützung mit eMedien« und als »Unterstützung mit eCommunications«.

Eine Konsequenz aus dieser Definition von E-Learning betrifft das Verhältnis zum Schlagwort »Blended Learning« (oft auch als »hybrides Lernen« [MS02b] oder »Lernen im Medienverbund« [BH04] bezeichnet, [Ben03]), das vor allem von der E-Learning-Industrie geprägt wurde. Ohne wissenschaftliche Fundierung bezeichnet es landläufig die Kombination der Nutzung von elektronischen Lernressourcen und traditionellen Präsenzlernformen zur wechselseitigen Bereicherung. Mit der in dieser Arbeit benutzten Definition von E-Learning ist damit ein »Blended Learning«-Ansatz ein spezieller E-Learning-Ansatz, was auch durch die Einschätzung gedeckt ist, dass bislang in den meisten Fällen rechnerunterstütztes Lernen als Ergänzung zu Präsenzlernelementen verstanden wurde, obwohl eine starke Technologieorientierung zu beobachten war.

Im angelsächsischen Sprachraum sowie im Rahmen der EU-Forschungsinitiativen hat sich in den letzten Jahren in Abgrenzung zum Marketing-Begriff »E-Learning« der Begriff »technology-enhanced learning« herausgebildet, der mit der obigen Definition von E-Learning im wesentlichen gleichzusetzen ist. Dabei weist »enhanced« darauf hin, dass nicht nur klassische Lernprozesse unterstützt, sondern auch neue Lernformen geschaffen werden.

Im Rahmen dieser Arbeit steht die Lernunterstützung mittels Informationsdiensten im Vordergrund, so dass eine Beschränkung auf E-Learning in der hier vorgestellten, weit gefassten Definition sinnvoll ist. Denn die Nutzung von Informationsdiensten impliziert hier immer, dass E-Learning vorliegt.

2.3. Thesen zum arbeitsbegleitenden Lernen

Was sind die wesentlichen Elemente und Charakteristika des arbeitsbegleitenden Lernens? Um diese herauszuarbeiten, werden im folgenden vier Thesen aufgestellt, die das Wesentliche zusammenfassen und die mit Hilfe der Literatur begründet werden .

- **These 1: Arbeitsbegleitendes Lernen hat viele Formen.** Arbeitsbegleitendes Lernen nimmt viele Formen an, die sowohl formelle als auch nicht-formelle oder informelle Lernformen umfassen. Ebenso existiert eine Vielzahl von didaktischen Strategien. Hier ist eine Grobstrukturierung erforderlich, die auf das arbeitsbegleitende Lernen und seinen betrieblichen Kontext abgestimmt ist.
- **These 2: Arbeitsbegleitende Lernprozesse sind nicht isoliert betrachtbar.** Im Gegensatz zum Lernen in künstlichen Umgebungen wie Schule oder Seminaren ist beim

arbeitsbegleitenden Lernen eine rein rezeptive Handlung des Lernenden nicht möglich bzw. nicht sinnvoll. In der Pädagogik betonen neuere Ansätze, dass Lernprozesse Artefakte erstellen, die anderen zur Unterstützung ihrer Lernprozesse dienen können. Dementsprechend reicht es nicht aus, das einzelne Individuum und seine Lernprozesse zu betrachten. Gerade in betrieblichen Umgebungen ist die überindividuelle Dimension von großer Bedeutung. Dazu gehören soziale, aber auch organisationale Beziehungen.

- **These 3: Arbeitsbegleitendes Lernen ist stark von der Situation beeinflusst.** Das arbeitsbegleitende Lernen kann nicht aus der Unternehmensumwelt herausgelöst werden, sondern muss andere, zum einen situationelle Faktoren, zum anderen überlagernde Unternehmensprozesse wie Wissensmanagement, Personalentwicklung (hier insbesondere das Kompetenzmanagement) und Geschäftsprozessmanagement (hier insbesondere Performance-Support), berücksichtigen.
- **These 4: Arbeitsbegleitendes Lernen braucht pädagogische Führung.** Bei vielen technologiegetriebenen Ansätzen wird oft ein wichtiger Aspekt vernachlässigt: der Übergang zwischen Lernen und Arbeiten ist nicht nur aus organisatorischer Sicht schwierig, sondern stellt auch für den Mitarbeiter eine kognitive Herausforderung dar, in der er sich Wissenslücken bewusst werden und diese in Lernaktivitäten umsetzen muss. Um diese Belastung in Grenzen zu halten und damit die Barrieren zu vermindern, muss auf das Konzept der pädagogischen Führung gesetzt werden, die den Lernenden vor Überforderung bewahrt.

2.4. Arbeitsbegleitendes Lernen hat viele Formen

Das Auffälligste bei der Beschäftigung mit arbeitsbegleitendem Lernen ist die Vielfalt an unterschiedlichen Lernformen und die daraus resultierende Vielfalt an unterstützenden Methoden. Diese Vielfalt macht eine ganzheitliche Beschäftigung mit der Thematik schwierig.

Während in der E-Learning-Diskussion (bzw. dem rechnerunterstützten Lernen als Vorläufer) sowie auch in der betrieblichen Bildung traditionell das formelle Lernen in Seminaren, Kursen etc. dominiert hat, treten in den letzten Jahren verstärkt »weniger formelle« Lernformen in den Vordergrund. Obwohl aus sprachlicher Sicht die Differenzierung in formell, informell und nicht-formell nicht sonderlich glücklich ist, hat sie sich (im wesentlichen durch Aktivitäten der UNO getrieben [SR02]), dennoch durchgesetzt, auch wenn immer wieder Kritik daran geäußert wird ([Str04], [Doh01], [SR02]). Aus diesem Grund bildet die Begrifflichkeit auch die Grundlage für diese Arbeit:²

²Eine ausführliche Diskussion der Begriffsgeschichte und der unterschiedlichen Definitionsversuche findet

- **Formelles Lernen** ist eine Lernform, die sich im wesentlichen institutionalisiert vollzieht. Die Inhalte sind stark vorstrukturiert. Oft führt formelles Lernen zu einem Abschluss oder einem Zertifikat.
- **Nicht-formelles Lernen** bezeichnet ein Lernen, das in Aktivitäten stattfindet, die nicht als Bildungsmaßnahme gedacht sind, das aber durch den Lernenden bewusst wahrgenommen und gesucht wird (intentionales Lernen [Ove01]).
- **Informelles Lernen** hingegen ist das Lernen in alltäglichen Situationen, das üblicherweise ungeplant ist und meist als zufällig wahrgenommen wird. Man spricht hier auch oft von »implizitem Lernen« [Deh02] oder »beiläufigem Lernen« [Kir04].

Bedeutsam ist die Unterscheidung der Lernformen insbesondere deshalb, weil die Lernunterstützung beim formellen Lernen im wesentlichen im »Vermitteln« (training) besteht, während für die beiden anderen Formen die Schaffung von Lernmöglichkeiten (oder der Abbau von Lernbarrieren) im Vordergrund steht [SR02]. Es wird auch oft betont, dass nicht-formelles Lernen im wesentlichen durch den Lernenden »nachgefragt« wird, während formelles Lernen »angeboten« wird [Ove01]. Arbeitsbegleitend kann hierbei beides geschehen: formell in Form von vorstrukturierten Kursen mit entsprechender Lernerfolgskontrolle, nicht-formell oder informell in Form von Kollaborationsumgebungen, selbstgesteuerter Suche nach geeigneten Lernressourcen etc.

2.5. Arbeitsbegleitende Lernprozesse sind nicht isoliert betrachtbar

Das traditionelle Bild bzgl. Fort- und Weiterbildung geht von einer starken Asymmetrie aus: der Lehrende vermittelt, der Lernende empfängt. Dementsprechend vollziehen sich auch Lernprozesse in diesem Paradigma im wesentlichen isoliert voneinander, d.h. es ist hinreichend, die Methoden der Lehrenden und den einzelnen Lernenden zu betrachten, allenfalls noch ergänzt um die Interaktion in einer Lerngruppe, die alle in der strukturell gleichen Lernsituation sind (z.B. Auszubildende, Teilnehmer in einem Seminar etc.). Mit der Individualisierung der Bildungsprozesse gerät dieses Modell allerdings an seine Grenzen, da mit der Vielzahl der Inhalte und Lernbedarfsarten keine strikte Rollentrennung mehr aufrechtzuerhalten ist, die ja eine Planbarkeit voraussetzt.

Dementsprechend betonen neuere Ansätze, die sich interessanterweise oft außerhalb der traditionellen Pädagogik entwickelt haben, die Wechselseitigkeit und den produktiven Charakter von Lernprozessen selbst, so z.B. der »Relate-Create-Donate«-Ansatz von SHNEIDERMAN [Shn98] (vgl. Abb. 2.1). Hierbei wird davon ausgegangen, dass Lernen im

sich in [SR02] und [Ove01]

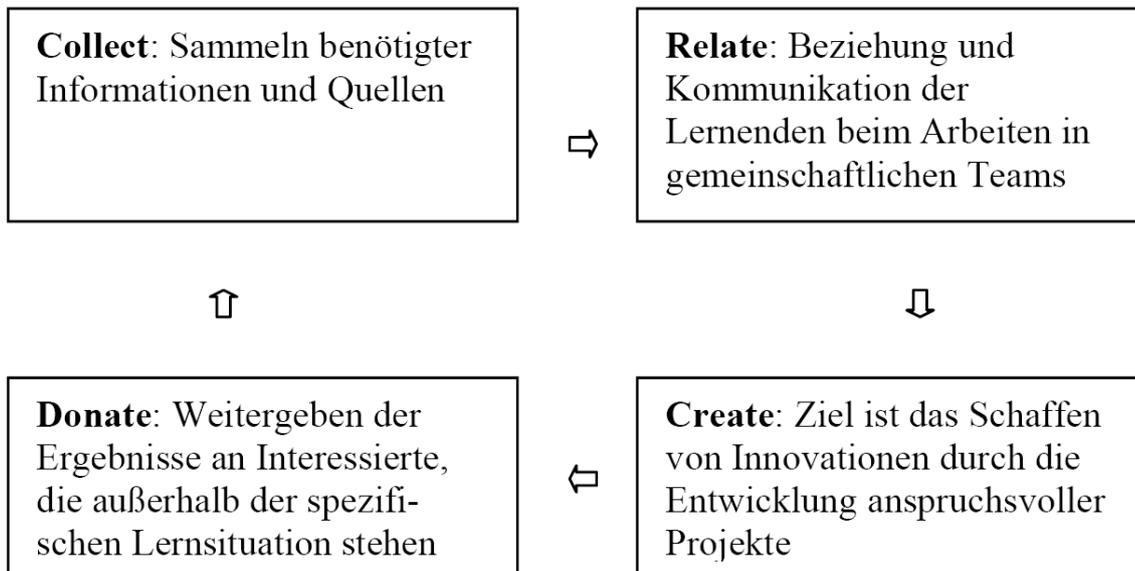


Abbildung 2.1.: Shneidermans Komponenten modernen Lernens [Shn02]

wesentlichen mit einem In-Beziehung-Setzen (*Relate*) von Bestehendem beginnt, das zuvor im Rahmen einer Sammlungsphase (*Collect*) zusammengetragen wurde (was Ressourcen und Menschen einschließt), dann überleitet zu einer schöpferischen Phase (*Create*), welche durchaus kollaborativ in einer Lerngruppe stattfinden kann. Das Ergebnis kann dann anderen zur Verfügung gestellt werden und als Basis für darauffolgende Lernprozesse dienen (*Donate*). Mit HAPKE kann man sagen: »Das Ergebnis zeitgemäßen Lernens ist im Optimum ein Wissensprodukt, das nicht nur für den Lernenden relevant ist.« [Hap05]. Hierbei ist nicht ausgeschlossen, dass es innerhalb dieses Prozesses Lehrende gibt, die auf der Basis der Ergebnisse ihrer Lernprozesse anderen bei der Wissenskonstruktion helfen, jedoch wird prinzipiell eine Peer-to-Peer-Philosophie vorausgesetzt, in der es zum einen keine Exklusivität gibt (i.d.R. gibt es mehrere Lehrende) und zum anderen die Rollenverteilung »Lehrender-Lernender« einen flüchtigen Charakter hat.

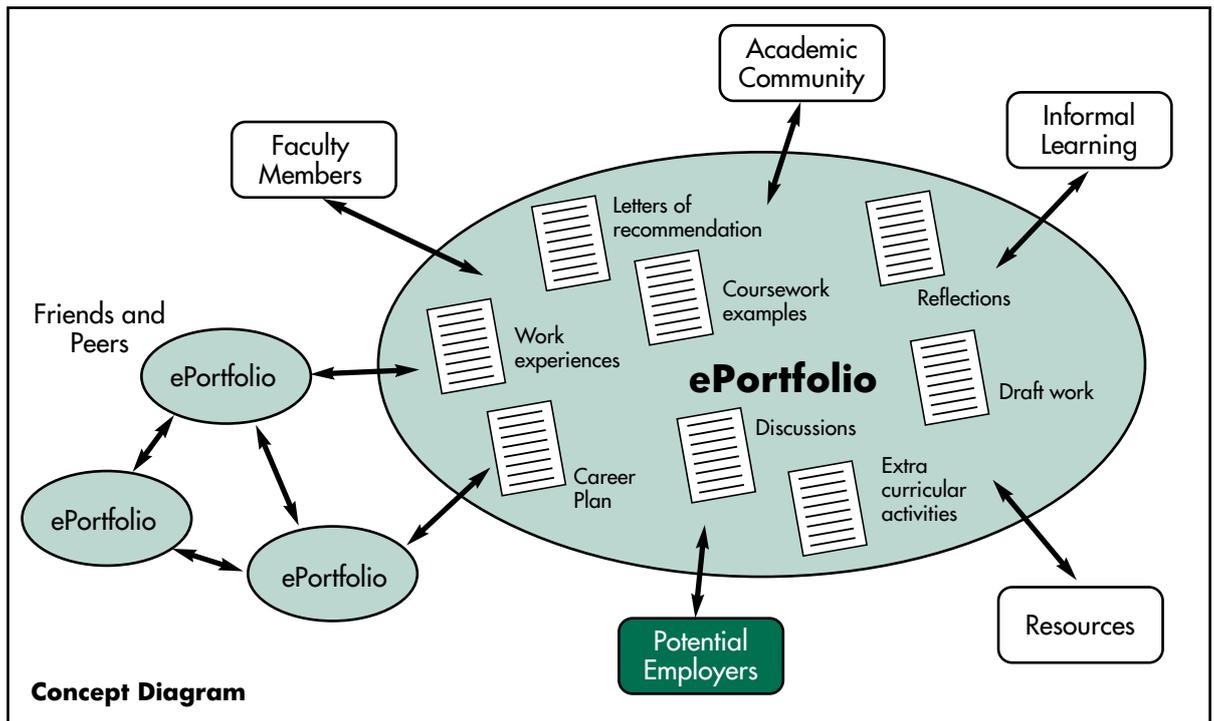
Ein Beispiel ist die Entwicklung im E-Learning-Bereich in Richtung sog. elektronischer Portfolios oder Weblogs. **Elektronische Portfolios** sind hierbei »elektronische Mappen« ([BC05], [Bar02]), welche im einfachsten Fall dokumentieren, was zu welchem Zeitpunkt, evtl. mit welchem Ergebnis »gelernt« wurde, um so als Bestandteil des beruflichen Lebenslaufes zu dienen, der auch für stärker individualisierte Lernprozesse geeignet ist (mit vielen extracurricularen Aktivitäten). Zusätzlich dienen sie auch für den Lernenden dazu, das Gelernte zu ordnen und strukturieren (insbesondere angefertigte Arbeiten o.ä. Artefakte) und darüber zu reflektieren. (vgl. Abb. 2.2). Sie können so auch die Basis für die Interakti-

cognitive absorption.

A learning landscape framework allows faculty and students to consider outcomes outwith the rigid structure of course outlines and requirements, incorporating a learning landscape, a combination of XML generalisations could be used to allow a user to transfer their portfolio between institutions, or perhaps for different learning systems to talk to each other. The result would be a standardised technology allowing students to converse with each other and learn

simple portfolios and establish connections to those created by their friends, business contacts and Web communities that interest them. This functionality could potentially be incorporated into an ePortfolio system allowing the user to share their uploaded work, as well as reflections about their

learning landscape, a combination of XML generalisations could be used to allow a user to transfer their portfolio between institutions, or perhaps for different learning systems to talk to each other. The result would be a standardised technology allowing students to converse with each other and learn



14

ISSUE 29 • OCTOBER 2004 Visit the LTSS website (www.ltss.bris.as.uk) for all your online resources

Abbildung 2.2.: Konzeptuelle Darstellung von elektronischen Portfolios [TW04]

on mit Tutoren oder Mentoren bilden, die in einer losen Form Lernfortschritte überwachen und gezielt Anstöße für die weitere Entwicklung geben können. Diese elektronischen Portfolios können teilweise oder ganz auch anderen zugänglich gemacht werden, um so als Grundlage für die Interaktion zwischen Lernenden zu dienen [TLFH05]. Dabei liegt der Schlüssel zum Erfolg solcher Portfolio-Ansätze klar in den konkreten Interaktionsformen mit dem Portfolio (sowohl durch den Lernenden, als auch mit dem Lehrenden).

Eine eher technologiegetriebene Entwicklung stellen »Weblogs« oder kurs »Blogs« dar [Röl05], die als sehr einfach zu bedienendes Web-Content-Management-System entstanden sind, um insbesondere persönliche Tagebücher im Netz zu pflegen. Zunehmend wird auch ihr pädagogisches Potential erkannt und untersucht ([Efi04], [Mos05], [PF05]) – gerade auch im Bereich des arbeitsbegleitenden Lernens. Hierbei dienen Weblogs für den jeweiligen Autor als Reflexionsinstrument, für die jeweiligen Leser können sie Lernprozesse anstoßen oder voranbringen, vor allem im Bereich des informellen Lernens. Weblogs können als Erweiterung und Ergänzung der »elektronischen Portfolios« verstanden werden, in denen auch formellere Lernformen repräsentiert werden können.

Unverkennbar an diesen Entwicklungen ist die zunehmende Wandlung der Rolle des Lernenden von einem rein Empfangenden hin zu einem Gebenden, was besonders für

arbeitsbegleitende Lernprozesse zutrifft, die oft auch ad hoc stattfinden. Als Konsequenz daraus können diese Lernprozesse allerdings nicht mehr isoliert betrachtet werden; die Verkettung von individuellen Lernprozessen ist hierfür zu bedeutsam. Denn wenn Mitarbeiter in einem beträchtlichen Teil der Fälle voneinander statt von spezialisierten »Trainern« lernen, dann müssen auch die Interdependenzen berücksichtigt werden.

2.6. Arbeitsbegleitendes Lernen braucht pädagogische Führung

Gerne wird im Zusammenhang mit der Integration von Arbeiten und Lernen vergessen, dass das Einleiten und Steuern von Lernprozessen durch den einzelnen Mitarbeiter kognitiv mindestens so anspruchsvoll ist wie der Aufbau von neuen Wissensstrukturen. Innerhalb der Arbeitssituation muss erkannt werden, dass ein Wissensbedarf besteht, dieser muss formuliert und in Aktionen umgesetzt werden. Und gerade in der initialen Lernphase wird der Mitarbeiter dann häufig mit Dingen konfrontiert, die er noch nicht versteht und insoweit nicht einordnen kann. Auf affektiver Seite wird dadurch das Gefühl der Unsicherheit, das ohnehin durch die Wissenslücke entstanden ist, noch weiter verstärkt [Kuh04]. Dieses Phänomen wurde besonders bei Mitarbeitern beobachtet, die neu in einem Bereich sind [Swe93]. Es kommt zum sog. »information overload«, einem Zustand, der durch die Konfrontation mit einer Informationsmenge zustandekommt, die die Informationsverarbeitungsfähigkeiten des einzelnen überschreiten, bzw. zum »cognitive overload«, der bei netzwerkartigen Informationsstrukturen wie z.B. Hypermedia durch die fehlende Linearisierung entsteht [Con87], [Ger97].

Die Antwort auf diese Probleme kann nur lauten, dass man auch beim arbeitsbegleitenden Lernen nicht auf pädagogische Führung verzichten kann. Anders als Informationssysteme oder Wissensmanagementsysteme müssen Dienste zur Unterstützung des arbeitsbegleitenden Lernens nicht nur Zugänge schaffen, sondern auch durch den durch sie erschlossenen Raum so führen, dass es zu keiner Überforderung kommt. Diese Führung darf allerdings nicht als die klassische Vorstellung von einem Lehrer missverstanden werden, der den Lernprozess für seine Schüler komplett vorstrukturiert und überwacht. Hier sind durchaus abgestufte Formen von Führung denkbar, die sich beispielsweise auch am Kompetenzniveau festmachen lassen, wie z.B. von RÖDER [Röd03] anhand der fünf Kompetenzniveaus von DREYFUS & DREYFUS [DD86] dargestellt:

- **Neuling.** Das Wissen wird am besten durch Anlernen und Üben (*drill and practice*) erworben, also durch sehr stark vorstrukturierte Lernprozesse.
- **Fortgeschritten.** Hierbei finden die Lernprozesse stärker selbstgesteuert statt, wobei Unterstützung durch Tutoren existiert. Diese führen durch gezielte Interventionen und stehen für Rückfragen zur Verfügung.

- **Kompetent.** Auf dieser Ebene ist es zum ersten Mal möglich, auf eine stark asymmetrische Rollenverteilung zu verzichten und auf kollaborative Lernumgebungen zu setzen, in denen Wissen gemeinsam erarbeitet und ausgetauscht wird. Die Führung ist hier eher in der Gruppendynamik zu sehen, wo wechselseitig gegenseitige Hilfen gegeben werden.
- **Gewandt.** Hier kann Lernen in komplexen Situationen mit eigenständiger Problemlösung stattfinden. Führung spielt in der Auswahl der Lernsituation und der Nachbereitung der Problemlösungsstrategien eine Rolle.
- **Experte.** Auf dieser Stufe findet Lernen in Form von völlig selbstgestalteter Forschungstätigkeit statt, und das Wissen wird mit Gleichgesinnten ausgetauscht. Führung existiert aufgrund der sehr offenen Lernsituationen praktisch nicht mehr.

Hierbei ist wichtig, dass es nicht pauschal ein Kompetenzniveau für einen Mitarbeiter gibt, sondern in unterschiedlichen Kompetenzbereichen die Niveaus auch unterschiedlich gelagert sein können.

Dieser Aspekt des arbeitsbegleitendes Lernen ähnelt den Problemen, die bei der Informationsrecherche auftreten. Aus diesem Grunde sollen die Probleme und die Möglichkeiten zur Überwindung aus der Forschung zum Informationsverhalten abgeleitet werden, was in einem separaten Kapitel dargestellt wird.

2.7. Arbeitsbegleitendes Lernen ist stark von der Situation beeinflusst

Das vielleicht kennzeichnendste Element des arbeitsbegleitenden Lernens ist die Unmittelbarkeit der Situation,

- durch die das Lernen motiviert ist
- in der der Lernerfolg überprüft werden kann

Diese Situation ist nichts anderes als die Arbeitssituation, die gekennzeichnet ist durch die Einordnung in Unternehmensprozesse und -strukturen, durch die bestehenden sozialen Netzwerke und den persönlichen Zustand (sowohl kognitiv als auch affektiv). Hierbei ist aus didaktischer Sicht bedeutsam, dass diese Arbeitssituation die authentische Umgebung ist, in der der Lernende später auch das Wissen anwenden soll und somit keine Trennung zwischen Erwerb und Anwendung besteht [Ste00]. Dies wird vor allem von der pädagogischen Richtung des **situierten Lernens** betont, die – üblicherweise auf das



Abbildung 2.3.: Zusammenhang von Unternehmensprozessen rund um das arbeitsbegleitende Lernen

schulische/akademische Lernen fokussiert – die Umgestaltung der künstlichen Lernsituation hin zu mehr Realitätsnähe fordert [DJ92]. Die didaktische Grundannahme ist dabei, dass je ähnlicher die Lernsituation zur späteren Anwendungssituation ist, desto größer ist auch der Transfererfolg [BS99]. Denn ohne diesen authentischen Bezug führen Lernprozesse häufig zu sog. »trägem Wissen« [BS85], das zwar scheinbar vorhanden ist, aber eben nicht zur Anwendung gebracht werden kann.

Neben der sozialen und persönlichen Dimension der Arbeitssituation ist der organisationale Aspekt derjenige, der für das arbeitsbegleitende Lernen besonders charakteristisch ist. Dies bezieht sich sowohl auf die Strukturen (Aufbauorganisation) als auch auf die Prozesse (Ablauforganisation). Besonders von Bedeutung ist dabei die Untersuchung der Wechselwirkung mit anderen Unternehmensprozessen, weshalb in diesem Abschnitt kurz die wesentlichen Prozesse beschrieben werden, die als organisationaler Kontext und Rahmen relevant sind: Personalentwicklung bzw. Kompetenz- und Skills-Management, Wissensmanagement und Geschäftsprozessmanagement (vgl. Abb. 2.3).

2.7.1. Arbeitsbegleitendes Lernen und Personalentwicklung

Die Aufgabe der Personalentwicklung ist die systematische, an den Unternehmenszielen ausgerichtete Gestaltung der benötigten Mitarbeiterressourcen. Ein wesentlicher Teil ist hierbei die Steuerung von Weiterbildungsprozessen von Mitarbeitern, so dass zukünftig benötigte Kompetenz auch tatsächlich bei den Mitarbeitern vorhanden ist, was heute üblicherweise mit den Begriffen Kompetenz- und Skills-Management bezeichnet wird.

Kompetenzmanagement bezeichnet den systematischen, an den übergeordneten Unternehmenszielen orientierten Umgang mit der Kompetenz des Unternehmens, was sich insbesondere in seinen Kernkompetenzen manifestiert. Kompetenzmanagement bewegt sich damit klar auf der Ebene der strategischen Unternehmensführung [BH05]. Dem gegenüber ist der Fokus von **Skills-Management** das Wissen und die Fähigkeiten der Mitarbeiter. Es beschäftigt sich mit einem systematischen Umgang mit Mitarbeiterkompetenzen bei der Personalauswahl, bei der Personalentwicklung und beim Personaleinsatz ([FBW91], [Aes04]). Dabei ist klar, dass erfolgreiches Kompetenzmanagement wohl nur schwer ohne eine Form von Skills-Management zu erreichen ist, so dass (trotz unbestreitbarer strategischer Anteile) Skills-Management im Wesentlichen als die operative Umsetzung begriffen werden kann.

Da Lernprozesse auf die Steigerung der Handlungskompetenz abzielen, stellen sie die Bausteine des Kompetenzmanagement dar. Daraus ergibt sich unmittelbar die Forderung nach einer Kompetenzorientierung der Steuerung dieser Lernprozesse: die Ziele müssen kompetenzorientiert beschrieben werden, damit eine Integration überhaupt ermöglicht wird. Allerdings operiert die Personalentwicklung üblicherweise auf einer größeren Granularitätsebene, so dass für den Zweck des arbeitsbegleitenden Lernens die Kompetenzmodelle verfeinert werden müssen.

2.7.2. Arbeitsbegleitendes Lernen und Wissensmanagement

Wie durch den Autor bereits in [Sch05a] dargelegt, ist das Ziel von E-Learning in betrieblichem Umfeld wie auch von **Wissensmanagement** die Unterstützung von Lernen im organisationalen Umfeld. Allerdings unterscheiden sich die Grundkonzeptionen der beiden Disziplinen: E-Learning geht wie das traditionelle Bildungswesen von einer starken Asymmetrie der Rollen Lehrender und Lernender aus. Aufgabe des Lehrenden ist die pädagogische Führung, die Unterstützung des individuellen Lernprozesses des einzelnen. Diese Führung kann wie bei klassischen Präsenzscenarien als »Unterrichten« verstanden werden, kann aber auch in der didaktischen Aufbereitung von Kursen liegen. Hier »führt« die Struktur und Reihenfolge der Inhalte. Beim Wissensmanagement hingegen ist die Rollentrennung schwach bis überhaupt nicht ausgeprägt. Es herrscht eher eine Peer-to-Peer-Philosophie vor, in der es darum geht, den Austausch zwischen Mitarbeitern zu fördern.

Eine pädagogische Führung ist nicht vorgesehen. Auf technischer Ebene konzentriert sich Wissensmanagement bislang im wesentlichen auf die Verbesserung der Informationssuche, indem neben der reinen Suche auch die Entstehung der Ressourcen und ihre Einordnung in den betrieblichen Kontext betrachtet wurde.

Präzisiert werden die Unterschiede zwischen Lernszenarien und der Nutzung von Wissensmanagementsystemen in [WZ04], was sich problemlos auf klassische Information-Retrieval-Systeme übertragen lässt. Gegenüber Lernszenarien fehlen die Festlegung von

- dem Lernort
- der Lehrperson
- den Lernzielen
- den Aufgaben (zur Vertiefung)
- den Prüfungen (zur Überprüfung des Gelernten)

Daraus folgt, dass Wissensmanagement sich eher darum kümmert, existierendes Wissen in unterschiedlichen Formen sichtbar zu machen. Dieses sichtbar gemachte Wissen kann als Basis für die Erstellung von Lernressourcen oder als Lernressource direkt genutzt werden. Hierbei ist zu berücksichtigen, dass der fehlende pädagogische Aspekt gezielt »nachgerüstet« werden muss. Bislang wird dieses Potential der Kopplung von Wissensmanagement und Lernmaterialerstellungsprozessen allerdings kaum genutzt.

2.7.3. Arbeitsbegleitendes Lernen und Geschäftsprozessmanagement

Das Geschäftsprozessmanagement gestaltet die Wertschöpfungsprozesse und die Unterstützungsprozesse im Unternehmen. Das Ziel ist hierbei die möglichst effiziente Ausführung. Berührungspunkte zum arbeitsbegleitenden Lernen ergeben sich an mehreren Stellen:

- Die Geschäftsprozesse bilden ein wesentliches Element der Arbeitssituation. Damit hat die Prozessgestaltung unmittelbare Auswirkung auf die Struktur der Arbeitssituation. Insbesondere lassen sich aus der Prozessgestaltung Anforderungen an die Kompetenz des Mitarbeiters ableiten, die Anhaltspunkte für den Lernbedarf des einzelnen liefern.
- Gleichzeitig sind die Geschäftsprozesse auch der Hintergrund für die Anwendung des erworbenen Wissens und die Entstehung von Artefakten, die im Rahmen des Wissensmanagements sichtbar gemacht werden.

Das geschäftsprozessorientierte Wissensmanagement hat hier bereits Methoden erarbeitet, wie sich prozessunterstützende Informationen direkt in die Prozesse einbetten lässt [AHMM02]. Dieser Ansatz ist zu verallgemeinern auf Lernunterstützung (anstatt nur Informationsbereitstellung).

2.8. Modelle

Als erster Schritt für die Erarbeitung von lernunterstützenden Diensten werden Modelle benötigt, die ein abstrahiertes Grundverständnis repräsentieren, das die Grundlage für die Konzeption bilden kann. Hierbei sind zwei Arten von Modellen notwendig:

- ein Mikromodell als Erklärungsmodell, »wie Lernen funktioniert« und wie es sich dementsprechend optimal unterstützen lässt
- ein Makromodell als Systematisierung von Lernformen und zum In-Beziehung-Setzen von einzelnen Lernprozessen miteinander und mit der Situation, in der sie sich vollziehen

Im folgenden Abschnitt wird der Konstruktivismus als geeignete Lerntheorie identifiziert und eine Sichtweise auf Lernumgebungen aus konstruktivistischer Sicht identifiziert. Im darauffolgenden Abschnitt wird skizziert, wie ein Makromodell aussehen sollte, bevor dieses im nächsten Kapitel erarbeitet wird.

2.8.1. Konstruktivismus als Lerntheorie

Überblick über Lerntheorien

Lerntheorien als Erklärungsmodelle, wie Lernen »funktioniert«, lassen sich in drei Großfamilien aufteilen:[Blu98]

- Der **Behaviourismus** geht davon aus, dass Lernen im wesentlichen ein deterministischer Prozess ist, der sich mittels Reiz-Reaktionsmustern beschreiben lässt. Der Behaviorismus bildet die Grundlage für klassische »drill & practice«-Lernprogramme und Intelligente Tutorielle Systeme.
- Der **Kognitivismus** beschäftigt sich mit dem kognitiven Prozessen beim Erfassen und Verarbeiten von Wissen. Hierbei werden nicht Inhalte komplett vorstrukturiert, sondern der Lernende muss Informationen entdecken und verarbeiten. Ziel ist hierbei die Steigerung der Problemlösefähigkeit.

- Der **Konstruktivismus** ist eine Weiterentwicklung des Kognitivismus und betont die aktive Rolle des Lernenden, der sich in möglichst realitätsnahen Szenarien das Wissen erarbeiten muss. Hierbei wird auch die Bedeutung des sozialen Kontextes betont.

Nachdem in den Anfangsjahren des E-Learning behaviouristische Ansätze dominierten, ist in den letzten Jahren stärker der Konstruktivismus in den Mittelpunkt des Interesses gerückt, da seine Vorstellungen und Konzepte besser geeignet für das »Lernen bei Bedarf« angesehen werden. Allerdings ist die Verankerung im Konstruktivismus in vielen Lernsystemen nur ein Lippenbekenntnis, was auch daran liegt, dass konstruktivistische Lernumgebungen durch ihre Freiheitsgrade deutlich schwieriger zu realisieren sind, insbesondere durch ihre Forderung nach Realitätsnähe. Dennoch erscheint der Konstruktivismus für das arbeitsbegleitende Lernen ein sehr guter Ausgangspunkt zu sein, weshalb diese Lerntheorie im folgenden auch noch näher analysiert wird.

Grundlage der konstruktivistischen Lerntheorie sind zwei Postulate:

- Jeder einzelne baut sich seine eigene innere Welt aus den äußeren Einflüssen auf.
- Dieses Aufbauen (oder Lernen) ist ein ganzheitlicher Prozeß, der Denken, Fühlen und Handeln umfaßt.

Demnach ist die Interpretation von Information kein passiver Wahrnehmungsprozeß, sondern stets ein kreativer Prozeß, der dem Einzelnen den Einbau der vorgefundenen Information in die eigenen kognitiven Strukturen abverlangt. Ein und dieselbe Information kann so auf höchst unterschiedliche Weise Aufnahme finden, abhängig vom »internen Kontext«, welcher sich von Individuum zu Individuum, aber auch zu unterschiedlichen Zeitpunkten bei ein und demselben Individuum unterscheidet.

Konstruktivistische Lernumgebungen

Die klassische Forschung rund um konstruktivistische Lernumgebungen konzentriert sich primär auf das schulische und das universitäre Lernen. Insofern wird dort immer vorausgesetzt, dass eine künstliche Lernsituation vorherrscht, die es abzumildern gilt. Dort erscheint die Forderung nach Problemorientierung in einem anderen Licht als im arbeitsbegleitenden Lernen, wo oft der Problemkontext unmittelbar ist. Umgekehrt ist oft die Forderung nach Führung des Lernprozesses schwieriger umzusetzen, so dass die Vorzeichen etwas anders sind. Deshalb soll im folgenden anhand des Modells für konstruktivistische Lernumgebungen von JONASSEN herausgearbeitet werden, welche Ansatzpunkte sich für die Lernunterstützung des arbeitsbegleitenden Lernens ergeben.

Das Modell (vgl. Abb. 2.4) ist in konzentrischen Kreisen aufgebaut, die die unterschiedlichen konzeptionellen Schichten repräsentieren. Der Kern (1) ist das Problem, das die

Grundlage für den Lernprozess darstellt. Um den Problemlösungsprozess zu unterstützen, werden auf den äußeren Ebenen folgende Unterstützungsformen angeboten:

2. **Verwandte Fälle.** Hier werden verwandte Probleme und ihre Lösungen bereitgestellt in Form von Fällen. Dies bietet in der betrieblichen Umwelt einen natürlichen Ansatz, der auch bereits mit fallbasierten Systemen im Bereich des Wissensmanagements abgedeckt wird.
3. **Informationsressourcen.** Hier werden Möglichkeiten zur Informationssuche bereitgestellt. Daraus läßt sich bereits folgern, dass die Informationssuche einen wichtigen Bestandteil von Lernprozessen darstellt, was weiter unten noch näher thematisiert werden soll.
4. **Kognitive Werkzeuge.** Hier werden Werkzeuge zur Verarbeitung und Strukturierung der gefundenen Informationen bereitgestellt, also alles, was den kognitiven Prozess verbessert.
5. **Kommunikations- und Kollaborationswerkzeuge.** Hier werden Möglichkeiten zur Interaktion mit anderen Lernenden und zur Zusammenarbeit bereitgestellt, die die soziale Dimension des Lernens repräsentieren.
6. **Soziale und kontextuelle Unterstützung.** Auf dieser Ebene kann externe Unterstützung in Anspruch genommen werden.

JONASSEN identifiziert dabei drei Unterrichtsstrategien (vgl. auch [Lit04], [Ste00]):

- A. **Modeling.** Die kognitiven Prozesse und Lernaktivitäten werden hierbei stark vordstrukturiert und im wesentlichen auf die Imitation durch Lernende gesetzt. Die Lösung für ein Problem wird exemplarisch vorgeführt, wobei die einzelnen Schritte und ihre Zusammenhänge erläutert werden.
- B. **Coaching.** Bei dieser Form führt der Lernende die Problemlösung selbst aus, und ein »Coach« kann Hinweise und Rückmeldungen an den Lernenden geben, den er aufmerksam beobachtet. Mit wachsender Handlungsfähigkeit des Lernenden wird die Unterstützung über die Zeit hinweg zurückgenommen.
- C. **Scaffolding.** Hierbei wird dem Lernenden ein Gerüst gebaut, damit er über seine derzeitigen Möglichkeiten hinaus die Wissenskonstruktion fortsetzen kann. Hierbei wird kontextabhängig in den Lernprozess eingegriffen, indem beispielsweise bestimmte Leitfragen gestellt werden oder Teile der Problemlösung herausgegeben werden, um die Lernenden aus einer Sackgasse herauszuführen. Die Aufgabebearbeitung erfolgt in kooperativer Form zwischen Lernendem und Experten.

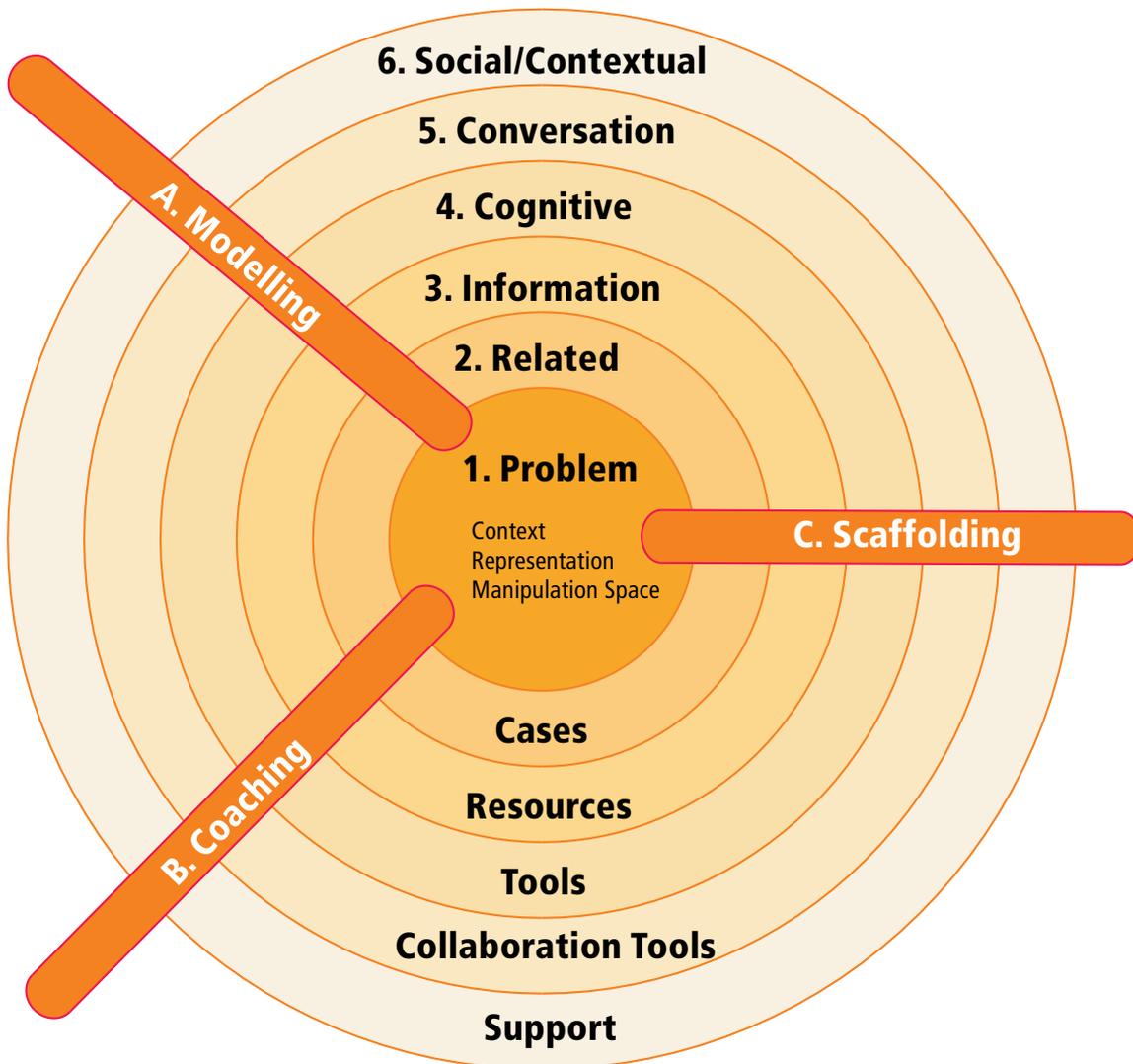


Abbildung 2.4.: Konstruktivistische Lernumgebungen nach Jonassen [Jon99]

Im Gegensatz zu traditionellen Unterrichtsmethoden tritt die reine Wissensvermittlung durch »Frontalunterricht« oder durch passives »Lesen« in den Hintergrund. Im Vordergrund steht das Lösen von authentischen Problemen und die situative Intervention und Hilfestellung durch einen begleitenden Experten.

2.8.2. Modell für die Beziehung zwischen individuellen Lernprozessen und den Artefakten

Mit der konstruktivistischen Lerntheorie ist ein gutes Orientierungsmodell identifiziert, wie Lernen im Kleinen funktioniert. Sie betont die aktive Konstruktion durch den Lernenden und die Bedeutung von authentischen Lernsituationen. Allerdings trifft sie nur wenig Aussagen über die Anatomie von solch authentischen Situationen mit ihrer organisationalen, sozialen und persönlichen Dimension. Insbesondere die organisationale und soziale Dimension erfordert es, nicht nur die Mikroperspektive zu betrachten, sondern auch makroskopisch auf den größeren Zusammenhang zu schauen:

- Welcher Zusammenhang besteht zwischen den Informationsartefakten, die als Nebenprodukt von kreativen Lernprozessen entstehen, und den Lernprozessen anderer?
- Wie lassen sich die unterschiedlichen lernbezogenen Unternehmensprozesse zusammenbringen?
- Wo kann ein System durch die Kenntnis der Arbeitssituation des Mitarbeiters lernunterstützend eingreifen?

Für diese Fragestellungen existieren weit weniger geeignete Modelle, so dass im folgenden Kapitel ein solches Modell auf der Basis von existierenden Ansätzen erarbeitet wird. Hierbei soll neben dem generellen Ziel einer Strukturierung der Lernformenlandschaft das Modell konkret folgende Anforderungen und Rahmenbedingungen erfüllen, die sich aus den vorangegangenen Thesen und Ausführungen ergeben (vgl. Abb. 2.5):

1. **Berücksichtigung formeller, nicht-formeller und informeller Lernformen.** Das Modell darf nicht allein auf organisierte Lernformen beschränkt sein, sondern muss gleichermaßen auch weniger formelle enthalten und sollte Aussagen darüber treffen lassen, wann welche Lernformen sinnvoll sind.
2. **Konstruktivistische Sichtweise.** Wie im vorangegangenen Abschnitt deutlich wurde, ist die konstruktivistische Sichtweise angemessen für das arbeitsbegleitende Lernen. Aus diesem Grund sollte der Konstruktivismus den lerntheoretischen Rahmen bieten.

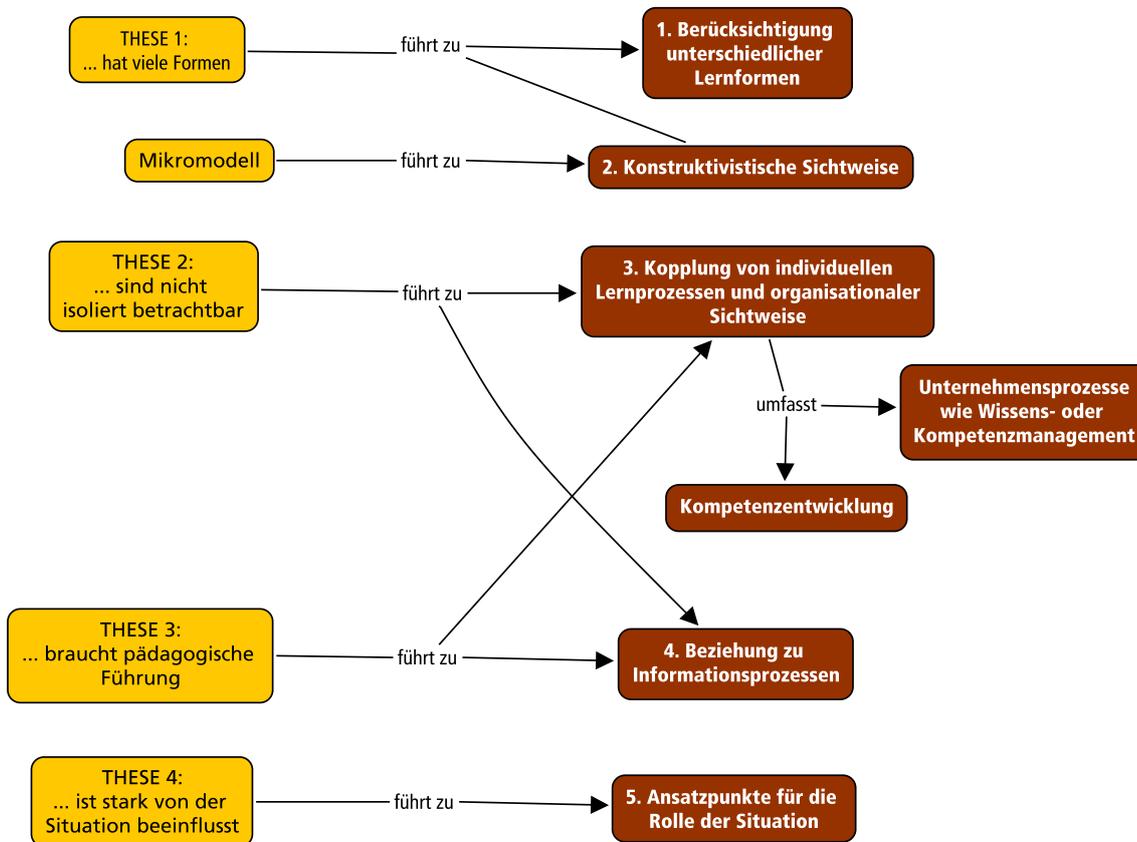


Abbildung 2.5.: Herleitung der Anforderungen aus den Thesen zum arbeitsbegleitenden Lernen

3. **Kopplung von individuellen Lernprozessen und organisationaler Sicht.** Für arbeitsbegleitendes Lernen ist die organisationale Sicht (was ist Ziel und Zweck des Lernens, wie soll Lernen geführt werden) von ebenso großer Bedeutung wie die individuelle Sicht (das Lernen an sich). Das Modell sollte diese beiden Sichtweisen miteinander verbinden, so dass ihr Zusammenwirken klar wird. Dies gilt insbesondere für die Beziehung zu anderen Unternehmensprozessen wie Wissensmanagement oder Personalentwicklung/Kompetenzmanagement, aber auch für die Beziehung zu individueller Kompetenzentwicklung.
4. **Beziehung zwischen individuellen Lernprozessen und Informationsprozessen.** Gerade in nicht-formellen und informellen Lernprozessen vollzieht sich Lernen in Informationsprozessen und über Informationsartefakte. Hierzu ist nicht nur die Nutzung solcher Artefakte zu betrachten, sondern auch ihre Erstellung.

5. **Ansatzpunkt für die Rolle der Situation in diesem Prozess.** Das Modell sollte auch Ansatzpunkte identifizieren, wo sich mit Hilfe von situationsbewussten Diensten Verbesserungen erzielen lassen. Es sollte also den Einfluss der Situation auf die individuellen Lernprozesse und die Kopplung von Lernprozessen beschreiben, so dass sich Interventionspunkte lokalisieren lassen.

Im folgenden Kapitel soll nun ein solches Makromodell, das diesen Anforderung gerecht wird, entwickelt werden.

3.

Der Wissensreifungsprozess als Makromodell

Ziel dieses Kapitels ist die Vorstellung eines Makromodells für das (recht weite) Feld des arbeitsbegleitenden Lernens. Dieses Modell dient im wesentlichen zur Strukturierung der Thematik, um in diesem Modell die unterschiedlichen Lernformen und insbesondere auch den Fokus der unterschiedlichen Disziplinen, die sich mit betrieblichem Lernen beschäftigen, zu verorten und eine leichtere Einordnung der im verbleibenden Teil erarbeiteten Lösungen zu ermöglichen. Zudem läßt sich anhand des Modells auch analysieren, an welchen Stellen derzeit ungelöste Probleme in der betrieblichen Praxis (und der Konzeptionen von Wissensmanagement, Personalentwicklung und E-Learning) liegen.

Dieses Makromodell liefert sowohl das Vokabular als auch den Bezugsrahmen für die weiteren Kapitel.

3.1. Einführung

Eine der größten Herausforderungen für Unternehmen ist derzeit die Bewältigung der gestiegenen Veränderungsdynamik, die sich in verkürzten Innovations- und Produktzyklen beobachtet werden kann. Hierbei wird oft genug das Wissen der Mitarbeiter zum Flaschenhals: der Fluss des »Wissens« von Forschung und Entwicklung bis hin zu Vertrieb und Kundenunterstützung ist weit von seinem Optimum entfernt – gerade weil Lernprozesse weitaus komplexer als Geschäftsprozesse und weniger zugänglich für die bewusste Gestaltung sind. Zwar hat man versucht, durch den Einsatz von E-Learning-Plattformen und Wissensmanagementwerkzeugen konventionelle Personalentwicklungsmethoden zu ergänzen, doch hat sich gezeigt, dass durch den jeweils anderen konzeptionellen Hintergrund und das jeweils unterschiedliche Verständnis von Lernen eine fragmentierte Lernlandschaft entstanden ist, die durch zahlreiche Brüche gekennzeichnet ist. Dies liegt nicht

zuletzt darin begründet, dass das arbeitsbegleitende Lernen ein weites Feld mit vielgestaltigen Lernformen ist. Es fehlt ein Modell, das diese Formen ordnet, den Wissensfluss und die ihn behindernden Brüche erklärt und Ansatzpunkte für die Überwindung dieser Brüche liefert.

3.2. Exkurs: Wissensbegriff

Wissen ist ein zentraler Begriff in diesem Kapitel. Es soll zwar hier nicht die philosophische Diskussion um den Begriff »Wissen« zusammengefasst oder fortgesetzt werden. Stattdessen geht es aber darum darzulegen, welche Grundkonzeption gemeint ist, wenn im Rahmen der Arbeit von Wissen gesprochen wird. Im Rahmen der Beschäftigung mit dem Wissensmanagement wurde klar, dass das Verständnis von dem, was Wissen darstellt, einen entscheidenden Einfluss auf die entwickelten Lösungen und die darin verwendeten sprachlichen Metaphern hat.

3.2.1. Wissensbegriff dieser Arbeit

Als Grundalternativen für einen Wissensbegriff haben sich im Umfeld von organisationalem Lernen, Wissensmanagement und E-Learning die folgenden herausgebildet:

- **Wissen als individuelle Konstruktion vs. Wissen als objektivierbare Größe.** Diese beiden Grundalternativen korrespondieren mit den Traditionen des Konstruktivismus und des Objektivismus [Ste02]. Während der Konstruktivismus im wesentlichen davon ausgeht, dass Wissen nur im Menschen existieren kann und dementsprechend nicht direkt bzw. in seiner radikalen Ausprägung überhaupt nicht objektiviert werden kann, sieht der Objektivismus keine Probleme darin, den Inhalt von Büchern o.ä. als Wissen zu bezeichnen. Vielfach wird sogar davon ausgegangen, dass objektiviertes Wissen das eigentliche Ziel sein sollte. Während in den ersten Phasen des Wissensmanagement der Objektivismus dominierte (z.B. [PRR03]), stehen nun stärker durch den Konstruktivismus bestimmte Konzepte im Vordergrund (z.B. [BS02a], vgl. [Sno02]), was letztlich auch den Siegeszug des Konstruktivismus im E-Learning-Bereich widerspiegelt.
- **Wissen als Prozess vs. Wissen als Produkt/Zustand.** Diese beiden Grundalternativen sind weniger offensichtlich. Klassischerweise herrschte der statische Wissensbegriff vor, in dem Wissen entweder als Zustand des menschlichen Geistes oder als »Produkt« verstanden wurde (vgl. [SZ95], [NT95], [PRR03]). Mit der Erkenntnis der Bedeutung von Kommunikations- und Veränderungsprozessen kamen auch Vorstellungen vom Prozesscharakter des Wissens auf. So versteht z.B. STACY Wissen

als einen aktiven Prozess des Inbeziehungsetzens [Sta01] oder BROWN & DUGUID als Prozess von Wissen und Handeln [BD00]. Eine gute Gegenüberstellung der beiden Sichtweisen findet sich in [Men00] und in [Abe04]. Auch wird von SNOWDEN [Sno02] oder auch im Münchner Wissensmanagementmodell [RR01] die Dualität beider Sichtweisen betont.

Klassisch für die Bestimmung des Wissensbegriffs im Umfeld der Informatik ist auch die Wissenshierarchie, die die Begriffe »Daten«, »Information« und »Wissen« gegeneinander abgrenzt [Nis02]. Hierbei visualisiert die Breite der Pyramide die Fülle an verfügbaren Einheiten (*abundance*), und die Höhe die Handlungsorientierung (*actionability*). Aus Daten wird Information, indem die Daten interpretiert werden, also in einen semantischen Kontext gestellt werden. Aus Informationen wird Wissen, indem sie in einen Kontext zu konkreten Handlungen gestellt werden.¹

Wenn man diese Hierarchie im Blickwinkel der dynamischen Flüsse betrachtet, so gelangt man zum Modell der gerichteten Wissenspyramide (s. Abb. 3.1, [Nis02]), das auf Konzepten von [Tuo99] aufbaut, das als Modell für Kommunikationsprozesse zur »Weitergabe« von Wissen dienen kann. Auf der Senderseite wird Wissen zu Information, indem beispielsweise etwas niedergeschrieben wird; man spricht auch von Wissensartikulation oder »elaboration« [AN95]. Dies wird zu Daten (Dokument), bevor diese Daten übermittelt werden und auf der anderen Seite den umgekehrten Weg nehmen, wo sie in die Wissensstrukturen des Empfängers eingebaut werden. Wissen ist also das Ergebnis der Verarbeitung von Informationen durch das menschliche Bewusstsein (vgl. [SN97], [PB98], [KT05]). Durch die zwischengeschalteten komplexen Transformationsschritte (Artikulations- bzw. Lernprozesse) ist das beim Empfänger aufgebaute Wissen i.d.R. nicht mit dem ursprünglichen Wissen identisch (vgl. auch [Wil01], [Sch03b]), auch wenn man meistens der Einfachheit halber davon ausgeht, dass eine hinreichende Übereinstimmung besteht.

Dennoch sollten diese Artikulations- und Lernprozesse nicht zu vereinfacht gesehen werden, da ansonsten wesentliche Elemente der Wirklichkeit ausgeblendet und die entsprechenden Lösungen daran vorbei entwickelt werden. Gerade Strömungen aus dem symbolischen Interaktionismus (SI) wie z.B. [Blu73] können hier helfen, die Komplexität besser zu verstehen. Hierbei kann man in obigem Modell die Informationsebene gleichsetzen mit den Symbolen des SI. Dann ergibt sich aus den Grundaussagen des SI, dass die Bedeutung dieser Symbole sich erst in der Interaktion entwickelt und eine erfolgreiche Verständigung voraussetzt, dass erfolgreich gemeinsame Bedeutungen von einer hinreichend

¹Die Dreiteilung ist zwar klassisch, die Interpretation der unterschiedlichen Ebenen variiert jedoch. STENMARK gibt hier einen recht guten Überblick in [Ste02]. So definiert z.B. [Sie05] die Begriffe Daten, Information, Wissen anders und fügt eine weitere Ebene Bedeutung hinzu, die dem hier vorgestellten Wissen entspricht. Interessant an dieser Stelle ist auch die »strukturgenetische« Sichtweise von SEILER UND REINMANN [SR04], die Informationen als eine Form von öffentlichem Wissen und Daten als formalisierte Information sehen.

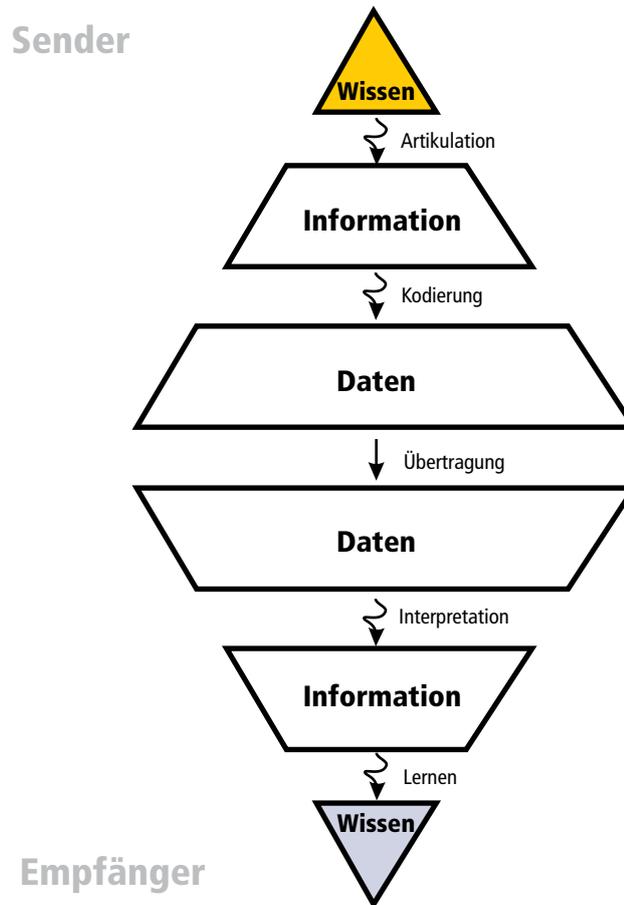


Abbildung 3.1.: Gerichtete Wissenspyramide für die Charakterisierung von Kommunikationsprozessen in Anlehnung an [Nis02]

großen Symbolmenge erarbeitet werden können (nach Mead [Mea75] »signifikante Symbole«). Ähnlich betonen aus hermeneutische Theorien wie z.B. das Encoding-Decoding-Modell von HALL [Hal80], dass zwar eine »präferierte Lesart« existiert (die im Idealfall der Intension des Senders entspricht), daneber aber auch eine vermittelnde, ausgehandelte und eine oppositionelle Lesart möglich sind. Dies verdeutlicht die Komplexität des Kodier-/Dekodierprozesses, so dass Abb. 3.1 nur als Ebenenmodell, nicht jedoch als Ablaufmodell zu verstehen ist.

ABECKER benutzt als pragmatische Arbeitsdefinition (aufbauend auf [Sch02a]) drei definitorische Elemente [Abe04]:

- Wissen ist zweck- und problemlösungsorientiert

- Wissen besteht aus vernetzten, kontextualisierten Informationen
- Wissen ist gebunden an die mentalen Modelle von Menschen

Fasst man die vorangegangenen Betrachtungsweisen von Wissen zusammen, so gelangt man zu folgenden Eckpunkten für eine Wissensdefinition, die die Grundlage für die vorliegende Arbeit bilden:

Definition 3.1 (Wissen) *Für diese Arbeit läßt sich der Wissensbegriff durch folgende beiden Aussagen umreißen:*

- *Wissen ist eine individuelle und subjektive Konstruktion, die durch (meist komplexe und oft soziale) Lernprozesse aufgebaut wird. Diese »Konstruktion« besteht aus dem Einordnen neuer Elemente in eine vorhandene Struktur und der Vernetzung vorhandener Elemente.*
- *Wissen wird erst dadurch vermittelbar, indem es als Information aufbereitet und objektiviert wird. Dieses Objektivieren bedingt und ermöglicht die Aushandlung eines gemeinsamen Verständnisses durch die Kommunikation zwischen Individuen.*

Diese Wissensdefinition hat zur Konsequenz, dass technische Systeme grundsätzlich nicht mit Wissen umgehen können, sondern nur mit Daten und — sofern sie die Semantik berücksichtigen — mit Informationen. Der Übergang von Wissen zu Information bzw. Information zu Wissen ist ein komplexer Prozess; ein einfaches Ablösen von Wissen vom Wissensträger ist nicht möglich (vgl. auch [BD00], [Bat05]).

3.2.2. Organisationales und Systemwissen

Es sei angemerkt, dass oft die Existenz eines **organisationalen Wissens** postuliert wird, das in Analogie zum individuellen Wissen bezeichnet wird. Hiermit wird das Phänomen beschrieben, dass in Prozessen, Organisationsstrukturen, Datenbeständen, aber auch in der Unternehmenskultur etwas enthalten ist, was zum einen eindeutig überindividuellen Charakter hat und zum anderen von entscheidender Bedeutung für das Unternehmen ist. Hier liegt es nahe, die Wissensmetapher zu benutzen.² Dementsprechend wird auch der Erwerb neuen organisationalen Wissens als organisationales Lernen bezeichnet. Auch wird in Analogie zur menschlichen Handlungskompetenz von den Kompetenzen eines Unternehmens gesprochen. Es sei nicht in Abrede gestellt, dass diese Phänomene eine große Bedeutung haben; allerdings verleitet die gleichlautende Bezeichnung zu Mißverständnissen, indem angenommen wird, dass individuelles Wissen und organisationales Wissen leicht ineinander übergehen können. Dem ist auch mit WILLKE [Wil01] entschieden zu widersprechen:

²Dass dies nicht zwingend ist, zeigt auch die Argumentation von REINMANN [Rei05], die bewusst kollektives Wissen als Information bezeichnet wissen will.

es handelt sich dabei um zwei voneinander getrennte Phänomene, zwischen denen der Übergang in hohem Maße komplex ist. In dieser Arbeit wird daher der Begriff des organisationalen Wissens so weit wie möglich vermieden, da es auch für die Problemstellung nicht notwendig erscheint.

Ähnlich verhält es sich auch mit der Metapher, dass Wissen in Rechnersystemen repräsentiert ist und dass das System selbst solches Wissen durch maschinelles Lernen aufbaut. Auch hier ist zu sagen, dass die Metapher zwar naheliegend ist, aber nicht unproblematisch, wenn man naiv das mit dem persönlichen Wissen gleichsetzt. Im Rahmen der Arbeit wird zwar durchaus von **Hintergrundwissen** gesprochen, jedoch sollte klar sein, dass die Beziehung zwischen Systemwissen und persönlichem Wissen dem eines Modells zu der von ihm erfassten Realität entspricht: es ist stark zweckorientiert und unvollkommen.

3.3. Existierende Modelle

In diesem Abschnitt sollen existierende Ansätze aus dem Wissensmanagement und dem Organisationalen Lernen analysiert werden, inwiefern sie sich für die Anforderungen in Abschnitt 2.8.2 eignen.

3.3.1. SECI-Modell

Das klassische Modell für betriebliche »Wissensprozesse« ist das sog. SECI-Modell von NONAKA UND TAKEUCHI [NT95]. Grundlage für die Beschreibung dieser Prozesse sind die beiden Wissensarten »implizites« und »explizites« Wissen. Anders als ursprünglich POLANYI [Pol67], auf den die Unterscheidung in explizites und implizites (*tacit*) Wissen zurückgeht, gehen NONAKA & TAKEUCHI grundsätzlich davon aus, dass sich implizites Wissen explizit machen läßt [SG03] und definieren so vier verschiedene Formen der »Wissensumwandlung« zwischen diesen Wissensarten (Sozialisierung, Externalisierung, Combination und Internalisierung) (vgl. Abb. 3.2):

- **Sozialisierung.** Bei der Sozialisierung wird implizites Wissen durch soziale Interaktion weitergegeben, wobei es bei den Empfängern wieder implizites Wissen darstellt.
- **Externalisierung.** Bei der Externalisierung wird aus implizitem Wissen explizites Wissen, was das Finden von Metaphern und deren Inbezugsetzung zu vorhandenem Wissen umfaßt.
- **Kombination.** Bei der Kombination werden aus unterschiedlichen expliziten Wissens-elementen neue explizite Wissensbausteine geschaffen.
- **Internalisierung.** Bei der Internalisierung wird explizites Wissen verinnerlicht und so zu implizitem Wissen.

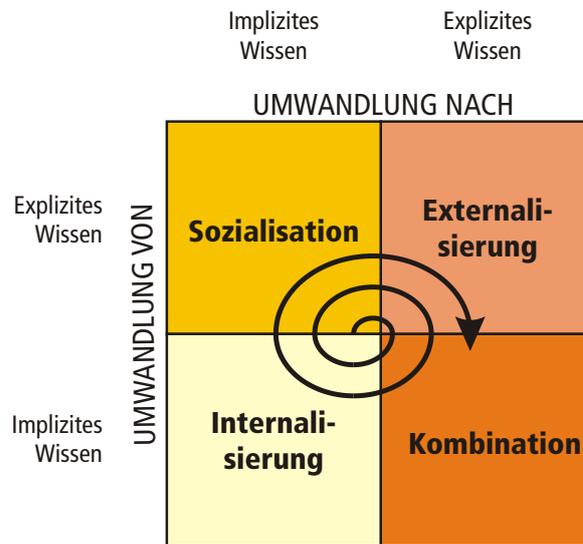


Abbildung 3.2.: Nonakas SECI-Modell nach [NT95], eigene Darstellung

Aufbauend auf diesen Wissenstransformationen beschreibt die sog. »Wissensspirale« (s. Abbildung 3.3), wie der wiederholte Prozess der Wissensschaffung und -transformation zu Innovation im Unternehmen führt. Hierzu unterscheiden NONAKA & TAKEUCHI zwei Dimensionen des Wissens: den Grad der Formalisierung (= epistemologische Dimension) und die Organisationsstufe (= ontologische Dimension). Die Wissensspirale entsteht, wenn man dem Wissensfluss über die Zeit hinweg folgt, wie es sich durch die wiederholten Transformationsphasen vom Individuellen hin zum Organisationalen und darüber hinaus entwickelt. Diese Transformationsphasen sind in starker Analogie zum SECI-Modell in fünf Phasen gegliedert: implizites Wissen austauschen (Sozialisierung), Konzepte schaffen (Externalisierung), Konzepte erklären (Kombination), Prototypen schaffen (ähnlich zu Internalisierung) und Wissen auswerten und übertragen. Im letzten Schritt findet hierbei eine Reflexion statt, aus der neue Einsichten generiert werden und die die nächste Transformationsphase initiiert.

Einer der wesentlichen Beiträge des SECI-Modells ist die Berücksichtigung der Dimension des impliziten Wissens und der impliziten Wissensvermittlung im organisationalen Kontext. Allerdings suggeriert die Wissensspirale einen eher harmonischen Wissensfluss, was im wesentlichen auf die etwas naive Sichtweise auf Lernprozesse (oder die Ausblendung der Komplexität der Prozesse) zurückzuführen ist. Auch wird davon ausgegangen, dass implizites Wissen grundsätzlich explizierbar ist, auch wenn der Aufwand hierfür einen unverhältnismäßigen Aufwand darstellt.

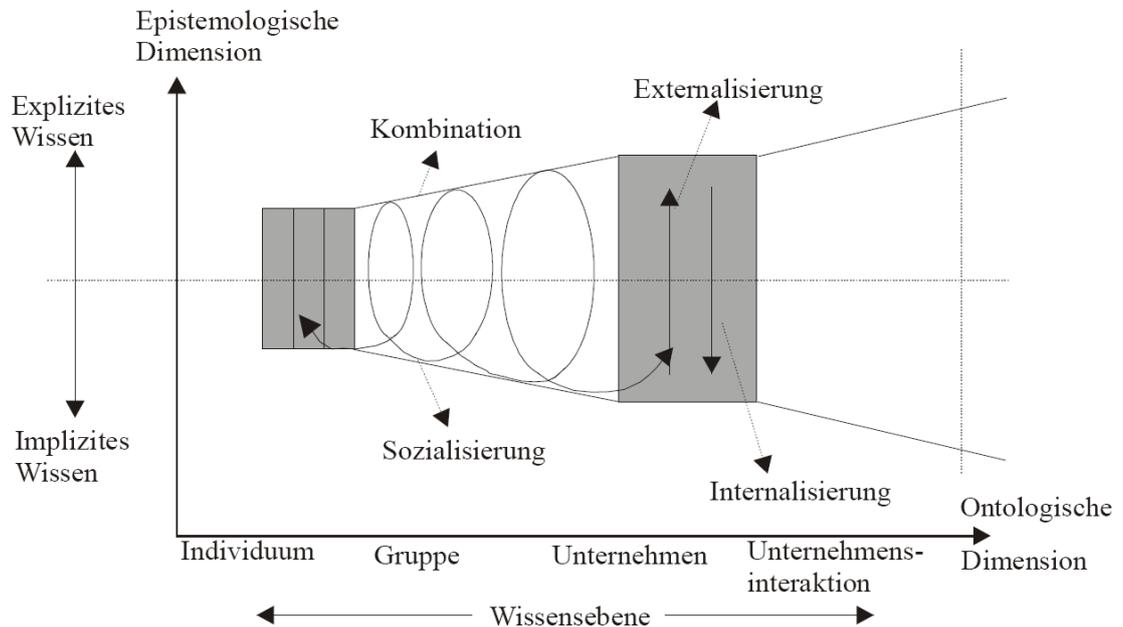


Abbildung 3.3.: Wissensspirale nach Nonaka [Non94]

Das Knowledge-Flow-Modell von NISSEN ([Nis02], [Nis02]) fügt dem SECI-Modell von NONAKA UND TAKEUCHI eine Zeitdimension hinzu, so dass sich eine Beziehung zum Wissensmanagementlebenszyklus herstellen lässt. Diese Beziehung bringt für die Fragestellung der Arbeit keine wirklichen neuen Erkenntnisse.

3.3.2. Snowdens Cynefin-Modell

Namensgebend für das Modell, das zur Strukturierung vorgefundener Phänomene von DAVID SNOWDEN entwickelt wurde ([Sno00], [Sno02], [KS03]), ist das walisische Wort »cynefin«, das sich nur schwer übersetzen lässt und in etwa soviel bedeutet wie »Lebensraum«, wobei das Bedeutungsspektrum schillert zwischen »Herkunft«, »Heimat« und »Ort der Lebensführung«.³ Es wurde entwickelt, um informelle und formelle Communities zu differenzieren und um sowohl mit strukturierten Prozessen wie auch unsicheren Bedingungen umzugehen. Dabei geht es primär um das »Sensemaking«-Verhalten [Ray03].

Das Cynefin-Modell ist ein zweidimensionales Modell. Die x-Achse stellt die (Austausch-) Kultur innerhalb der Organisation dar, wobei die beiden Pole Lernen und Lehren sind; die y-Achse klassifiziert die Sinngebung (*sense-making*) von geringer Abstraktion

³<http://www.cynefin.net/>

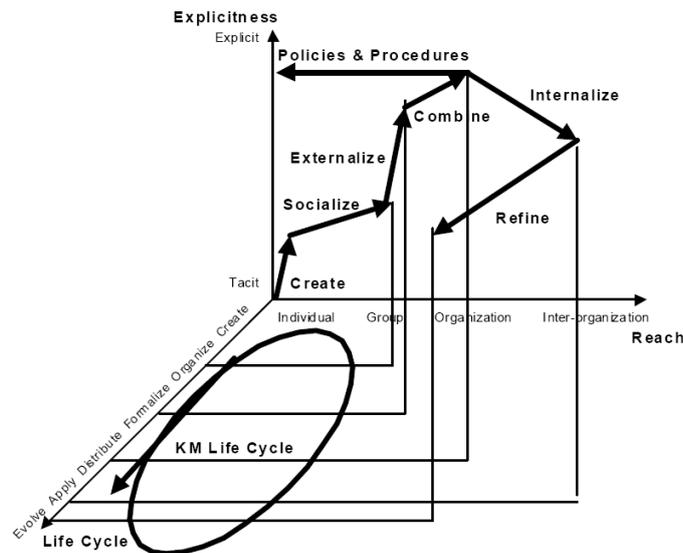


Abbildung 3.4.: Nissens erweitertes Knowledge-Flow-Modell [Nis02]

nach hoher Abstraktion. Hohe Abstraktion bedeutet dabei, dass Wissen sehr stark im Kontext verhaftet ist und somit nur Spezialisten zugänglich ist, während eine geringe Abstraktion etwas allgemein Zugängliches charakterisiert. Darauf aufbauend lässt sich der Wissensfluss wie folgt charakterisieren (vgl. Abb. 3.5):

- **Known.** In diesem durch Ordnung gekennzeichneten Bereich sind Ursache-Wirkungs-Beziehungen klar erkennbar, so dass Vorhersagemodelle aufgestellt werden können. Dementsprechend geht es um Effizienz, so dass strukturierte Techniken im Vordergrund stehen. Typische Lerninstrumente sind hier sog. »Best Practices« und Gruppen mit großem Zusammenhalt, die mit ausreichend Informationen versorgt werden.
- **Knowable.** Auch dieser Bereich ist durch Ordnung gekennzeichnet. Allerdings sind die Ursache-Wirkungs-Beziehungen nicht vollständig bekannt, weil in vielen Fällen das Explizitmachen zu aufwendig ist. Dementsprechend stehen Methodiken im Vordergrund, die Ursache-Wirkungs-Beziehungen sichtbar machen. Dies ist typischerweise die Domäne von organisationalem Lernen und Systemdenken. Strukturierte Methoden können demgegenüber gefährlich sein, wenn ihre Voraussetzungen nicht immer wieder in Frage gestellt werden. Geeignete Lerninstrumente sind hier »Communities of Practice« mit klarer Zielsetzung, in denen der analytische Erkenntnisstil vorherrscht, der die Komplexität der vorgefundenen Realität auf Grundprinzipien

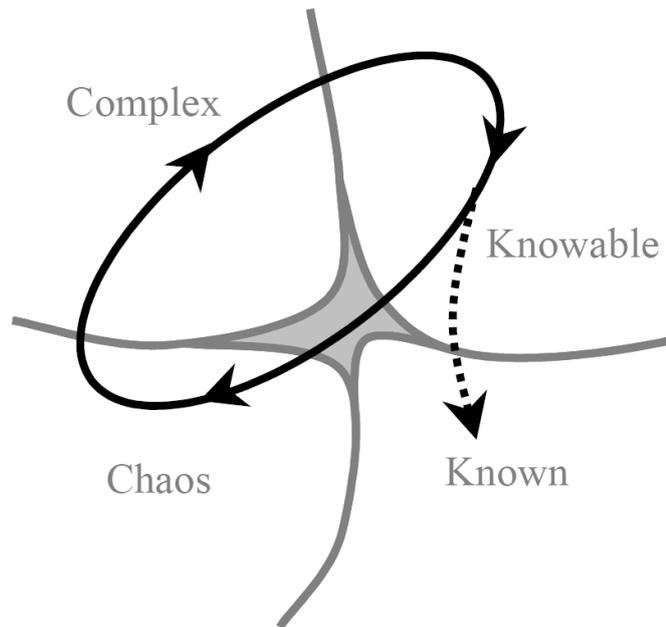


Abbildung 3.5.: Das Cynefin-Modell: Knowledge Flow

zu reduzieren versucht.

- **Complex.** Ursache-Wirkungs-Beziehungen sind aufgrund der Anzahl der Agenten und der Wechselwirkungen zwischen ihnen zu komplex, um sie mit analytischen oder kategorisierenden Verfahren zu behandeln. Mustererkennung liefert nur unvollkommene Resultate, da die Muster nicht stabil sind. Geeignete Lerninstrumente sind hier informelle Gruppen und soziale Netzwerke.
- **Chaos.** Dieser Bereich ist kognitiv der herausforderndste, da hier aufgrund der starken Dynamik und der zerfallenden Erkenntnisstrukturen keine Ursache-Wirkungs-Beziehungen ausgemacht werden können. Hier können nur temporäre informelle Gruppen helfen.

Das Modell geht über die Zeit hinweg davon aus, dass die einzelnen Bereiche zyklisch durchlaufen werden und es keinen stabilen Endzustand gibt. Es wird sogar postuliert, dass der Bereich »Known« eigentlich die Ausnahme darstellt und der größte und wichtigste Teil sich in den anderen drei Bereichen abspielt.

Das Cynefin-Modell ist besonders aufgrund seines Entstehungshintergrundes im Bereich der Communities sehr stark auf die soziale Dimension des Lernen konzentriert und kann so nicht das ganze Spektrum des nicht-formellen und formellen Lernens berücksich-

tigen. Allerdings sind im Cynefin-Modell wichtige Erkenntnisse enthalten, dass es »Wissensbereiche« gibt, in denen das klassische Lehr-Lern-Paradigma überhaupt nicht funktionieren kann, weil eben die stabilen Strukturen fehlen, die ein pädagogisches Aufbereiten erst möglich machen.

3.3.3. Lebenszyklus des Wissens

BIRKINSHAW postuliert in [BS02b] einen Lebenszyklus des Wissens, der eine marktorientierte Grundsicht hat und der sich in folgende Phasen aufteilt:

- *Schaffung von Wissen.* In dieser Phase »entsteht« das Wissen durch Innovationstätigkeit.
- *Mobilisierung des Wissens.* Durch Kodifizierung wird das Wissen ablösbar vom einzelnen Wissensträger.
- *Diffusion.* Hier kann das Wissen am Markt (also auch außerhalb der Organisation) verbreitet werden.
- *Commoditisation.* Hier wird das Wissen zum Gut, zum Produkt mit standardisierten Eigenschaften.

BIRKINSHAW ordnet unterschiedliche Wissensmanagementinstrumente den einzelnen Phasen zu. Dieses Modell ist allerdings zu sehr auf Wissen als etwas Vermarktbares ausgerichtet und lässt keine Rückschlüsse auf die einzelnen Lernprozesse zu. Insbesondere erscheint die Sichtweise vom »Wissen als Gut« als zu naiv, da hier – wie oben beim Exkurs über den Wissensbegriff skizziert – nicht klar zwischen dem persönlichen Wissen und einer organisationalen Abstraktion unterschieden wird.

3.3.4. Modell für organisationale Informationsverarbeitung nach Maier

MAIER hat ein Makromodell für die unterschiedlichen Arten der betrieblichen Informationsverarbeitung und ihre Zusammenhänge erarbeitet ([Mai04], vgl. Abb. 3.6). Die Wissensverarbeitung beginnt mit der Aufnahme externen Wissens aus der Umwelt des Unternehmens (1) oder der Identifikation von interessanten Wissensquellen innerhalb des Unternehmens bzw. der Organisation (2). Wissensquellen können dabei dokumentiertes Wissen, Personen, Gruppen oder andere Organisationseinheiten sein. Mit individuellem Lernen (3) werden die Wissensquellen in den Kern des Zyklus organisatorischen Lernens aufgenommen. Das Wissen wird mit anderen geteilt (4), so dass inter-subjektives Wissen entsteht. Mit Hilfe von Externalisierung, Formalisierung, Bewahrung sowie Legitimation entsteht

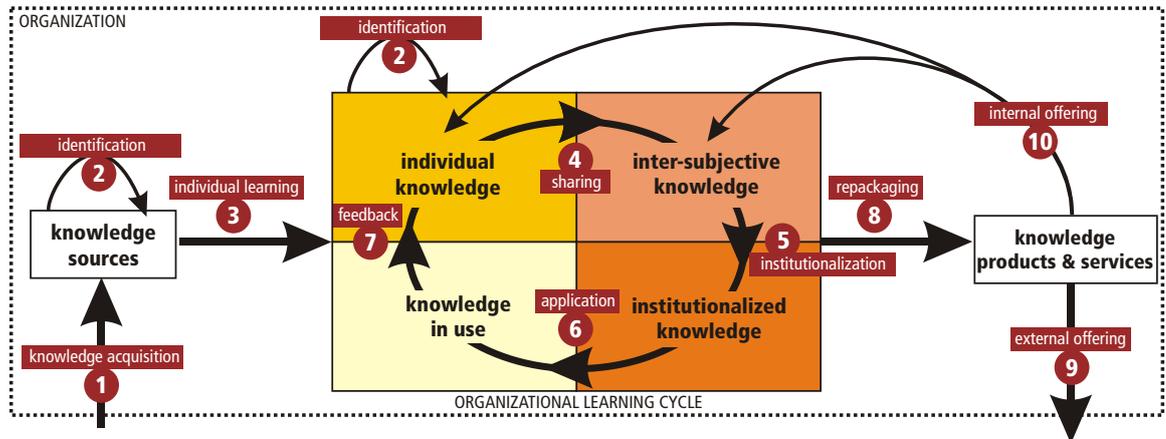


Abbildung 3.6.: Modell für die organisationale Informationsverarbeitung nach Maier [Mai04] (eigene Darstellung aus [MS07])

ORGANIZATIONAL LEARNING CYCLE

institutionalisiertes Wissen (5). Dieses repräsentiert formal und offiziell akzeptierte, legitimierte Praktiken, wie in gewissen Situationen reagiert werden soll (*espoused theories*) im Gegensatz zum tatsächlich genutzten Wissen (6, *theories-in-use*, [AS78]). Dieses kann, muss aber nicht kompatibel sein mit dem institutionalisierten Wissen. Die Resultate der Kombinationen und Wiederverwendung von Wissen, der getroffenen Entscheidungen und getätigten Handlungen gibt Feedback, das wiederum das individuelle Lernen anregt (7). Das im Zyklus organisatorischen Lernens geschaffene Wissen kann verfeinert und so aufbereitet werden (8), dass daraus Wissensprodukte und -dienste entstehen. Diese können an die Umwelt des Unternehmens abgegeben (9) oder wieder in den Zyklus eingebracht werden (10).

Dieses Modell bietet gegenüber den vorangegangenen Modellen bereits eine etwas detaillierte Sicht darauf, wie Informations- und Wissensflüsse in Unternehmen stattfinden. Insbesondere der Zyklus des organisationalen Lernens legt nahe, dass unterschiedliche Lernformen für unterschiedliche Abschnitte in diesem Zyklus existieren. Was allerdings in diesem Modell noch nicht zu erkennen ist, ist die Rolle die (Lern-)Situation. Ebenso erscheinen individuelle Lernprozesse undifferenziert als eine Phase in diesem Zyklus.

3.4. Wissensreifungsprozess: Grundidee und Phasen

Ausgangspunkt für das Modell des Wissensreifungsprozesses sind eine Reihe von Beobachtungen. Intuitiv ist es unstrittig, dass Ideen weitergegeben und weiterentwickelt werden und man am Ende immer noch den Bezug zum ursprünglichen »Wissen« erkennen

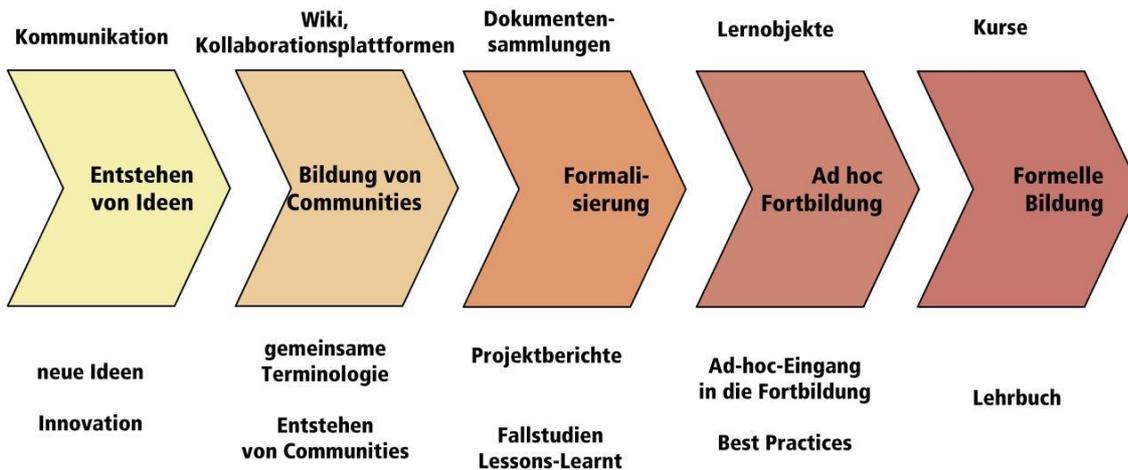


Abbildung 3.7.: Der Wissensreifungsprozess

kann, auch wenn dieser formal nicht unbedingt fassbar ist. So geschieht dies auch mit wissenschaftlichen Erkenntnissen, die sich mehr oder minder gut zurückverfolgen lassen. Offensichtlich gibt es – wenn man miteinander verbundene individuelle Lernprozesse betrachtet – eine Art Kontinuität. Dieses Phänomen rechtfertigt die Abstraktion in einen gedachten **Wissensfluss**, wobei das Fließen durchaus ein sehr komplexes Problemfeld sein kann.

Wenn wir weiterhin die makroskopische Entwicklung von Wissen betrachten, so finden wir viele Metaphern wie z.B. »konsolidieren« oder »in ein größeres Ganzes einordnen«, oder man spricht offen davon, dass etwas noch nicht »reif« genug ist. Dies deutet darauf hin, dass man von einer unterschiedlichen Qualität von Wissen sprechen kann. So ist man oft beispielsweise (noch) nicht in der Lage, etwas einem Laien zu vermitteln, weil man »es noch nicht ausreichend verstanden hat«, manchmal kann man eine neue Idee noch nicht einmal dem Kollegen vermitteln. Hier bietet sich die Metapher des **Reifens** an, das Veränderungen in der Vermittelbarkeit aufgrund einer veränderten Tiefe und Reichhaltigkeit des Verständnisses (im Sinne des Konnektivismus [Sie05]: einer besseren Vernetzung) charakterisiert.

Bringt man beide Phänomene zusammen, so ergibt sich, dass normalerweise die Reife entlang des Wissensflusses zunimmt. Je öfter ein bestimmtes Thema vermittelt wurde, desto besser wird es auch verstanden, desto leichter ist es auch, es wiederum weiterzuvermitteln.

Vor dem Hintergrund aktueller E-Learning- und Wissensmanagementansätze gelangt man zu folgenden fünf Phasen (vgl. Abb. 3.7):

- **Entstehen von Ideen.** In dieser Phase kommen neue Ideen auf und werden auf sehr informelle Art und Weise im Rahmen von Unterhaltungen und Diskussionen weitergegeben, ohne dass ein gemeinsames Vokabular existieren würde.
- **Bildung von Communities.** Mit der Entstehung von Communities vollzieht sich ein wichtiger Reifungsschritt: ein gemeinsames Vokabular entsteht, was eine zielgerichtete Zusammenarbeit an einem bestimmten Thema ermöglicht. Hier spielen Kollaborationswerkzeuge wie Wikis eine wichtige Rolle.
- **Formalisierung.** Wo in der vorigen Phase greifbare Artefakte erzeugt wurden, waren sie inhärent unstrukturiert wie z.B. Beiträge in Diskussionsforen, Blog-Einträge o.ä.. In dieser Phase werden bereits zweckorientierte, strukturierte Dokumente wie Projektberichte oder Entwurfsdokumente produziert.
- **Ad-Hoc-Fortbildung.** Die in der vorigen Phase produzierten Dokumente sind üblicherweise nur schlecht als Lernmaterialien geeignet, weil ihnen die didaktische Aufbereitung fehlt. In dieser Phase werden solche aufbereiteten Lernobjekte (oder auch Kurzfortbildungen im Präsenzstil) erstellt, so dass eine deutlich verbreitetere Zielgruppe in die Lage versetzt wird, sich das Wissen anzueignen.
- **Formelle Bildung.** Diese höchste Reifephase ordnet die Lernobjekte in einen größeren Zusammenhang ein und erstellt in sich geschlossene Kurse, so dass dieses konsolidierte Wissen auch Anfängern (z.B. im Rahmen der Lehre an Schulen oder Universitäten) vermittelt werden kann.

In Abb. 3.8 sind die »Eigenschaften des Wissens« und die geeigneten Lernformen detaillierter wiedergegeben. Interessant ist hierbei, dass gerade die Technologien, die derzeit für Lern- und Wissensmanagementzwecke populär sind (wie Wikis oder Weblogs), typische phasenübergreifende Technologien sind, die also die Brücke zwischen Phasen zu überwinden helfen, wobei allerdings dem Phänomen der Reifung in diesen Technologien bislang keine Beachtung geschenkt wurde.

3.5. Reifungsschritte und die Rolle der Situation

Es ist wichtig, dass man berücksichtigt, dass das Wissen stets in den Köpfen der Menschen existiert und die produzierten Artefakte (wie z.B. Textfragmente, Dokumente, Lernmaterialien) nur dazu dienen, dass andere besser in die Lage versetzt werden, die Thematik in ihre individuellen Wissensstrukturen einzubauen. Die Artefakte haben dabei eine gewisse Qualität, die teilweise ein Abbild der Reife des zugrunde liegenden Wissens darstellt.

	Eigenschaften des Wissens	Lernformen
Entstehung von Ideen	Wissen ist flüchtig und schwach vernetzt und kann dementsprechend nur schwer in kommunizierbare Information übersetzt werden. Ein gemeinsamer Kommunikationskontext ist zudem schwierig, da kein gemeinsames Vokabular existiert.	Soziale Lernformen sehr informeller Austausch
Bildung von Communities	Wissen wird zunehmend vernetzt, woraus sich auch innerhalb von schwach institutionalisierten Communities ein gemeinsames Vokabular herausbildet, das die Kommunikation (und wiederum die Vernetzung) vereinfacht.	Kollaboration (CSCW, CSCL) kooperative Wissenskonstruktion
Formalisierung	Durch den gestiegenen Vernetzungsgrad sind Formalisierungen in Dokumentenform möglich. Zwar liegen noch keine Erfahrungen in der effizienten Vermittlung vor, doch ermöglichen diese Formalisierungsbemühungen die Kommunikation ohne direkten Kontakt, ablösbar von den konkreten Personen.	Berichte Fallbasierte Lernformen selbstgesteuerte Lernformen Klassische Informationsrecherche
Ad-Hoc-Fortbildung	Auf dieser Stufe tritt vermehrt die Vermittlung für einen größeren Personenkreis in den Vordergrund. Das Wissen gewinnt eine gewisse zusätzliche Struktur durch didaktische Überlegungen (z.B. Praxisübungen). Für die Vermittlung wird allerdings noch vorausgesetzt, dass gewisse Grundstrukturen beim Lernenden vorhanden sind.	Lernobjekte Kurzschulungen Einmal-Seminare Best Practices
Formelle Bildung	Das Wissen wird in einen großen Gesamtzusammenhang eingeordnet, das einer Theoriebildung ähnelt. Hier ist ein größeres Themengebiet so strukturiert, dass auch Neulingen Informationen so präsentiert werden können, dass sie Grundstrukturen aufbauen können, so dass es sich für einen breiten Personenkreis eignet.	Standardisierte Schulungen Kurse Lehrbücher

Abbildung 3.8.: Eigenschaften der einzelnen Phasen des Wissensreifungsprozesses

Dementsprechend reift Wissen innerhalb dieses Prozesse nur dadurch, dass Menschen Informationen im Rahmen von Gesprächen, Diskussionen, aus Dokumenten oder Lernmaterialien aufnehmen (d.h. »lernen« (!)), indem sie diese aus ihrem ursprünglichen Entstehungssituation herauslösen und in ihr eigenes mentales Modell einbauen (vgl. Abb. 3.9). Durch die Beschäftigung mit der Thematik werden diese internen Wissensstrukturen transformiert, ergänzt und mit anderen verknüpft - Wissen »reift«. Bei einer bestimmten Gelegenheit wird dieses gereifte Wissen dann in ein Artefakt, das einen bestimmten, meist erweiterten Zielkontext besitzt, hinein übersetzt, z.B. durch das Einbringen in Diskussionsprozesse, die Erstellung von Dokumenten oder eben die Erstellung von Lernmaterialien.

Benutzt man die übliche Terminologie, so kann man sagen, dass sich die Artefakte auf der Informationsebene befinden. Der Wechsel von der Informations- auf die Wissensebene ist »lernen«; die umgekehrte Richtung wird oft als »Wissensartikulation« bezeichnet.

An dieser Stelle sei vor einer zu einfachen Sicht auf den Wissensreifungsprozess ge-

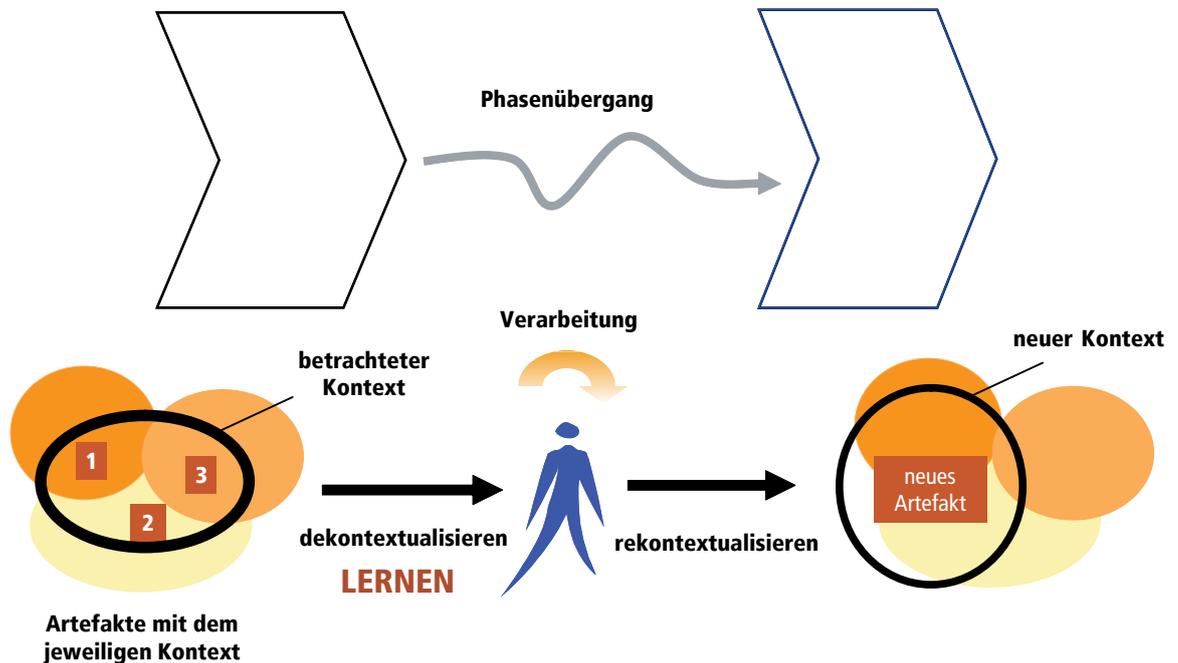


Abbildung 3.9.: Ein Reifungsschritt im Detail

warnen. Die streng lineare Darstellung, die zur Vereinfachung und auch aus dem Charakter eines beschreibenden Modells heraus gewählt wurde, spiegelt nicht die volle Komplexität wider. So ist die Kombination von vorhandenen Ressourcen nicht auf eine einzelne Reifestufe beschränkt, sondern kann beliebige Stufen umfassen. Der Reifegrad bestimmt sich dann nach dem dominierenden Element. Auch können sich Wissensflüsse aufspalten, bei denen Anteile abgespalten werden, die auf eine geringere Reifestufe zurückfallen.

3.6. Brüche innerhalb des Prozesses und Ansätze zu deren Überwindung

Wenn man den Prozess des Reifens entlang der o.g. Phasen nachvollzieht, so wird ersichtlich, dass innerhalb dieses Prozesses etliche Brüche auftreten, die zu suboptimalen Wissensflüssen im Unternehmen führen.

- **Wissenstransfer vs. Weiterbildung.** Im Wissensmanagement wird von einem selbstorganisierendem Prozess ausgegangen: jeder produziert und nutzt Inhalte: man teilt sie. Demgegenüber geht die Weiterbildung von einer klaren Trennung von Lehrenden (die didaktisch vorgebildet sind) und Lernenden aus. Dies führt unweigerlich

zu Konflikten zwischen den Systemen für die frühen und die späten Phasen des Prozesses.

- **Explizites Wissen vs. Lernressourcen.** Mit dem Konflikt der beiden o.g. Paradigmen ist auch das Problem verbunden, ob »explizites Wissen« ohne pädagogische Aufbereitung (also einfache Dokumente) als Lernressource taugt. Aus E-Learning-Sicht wird dies üblicherweise verneint, was dazu führt, dass die Produktion von Lernmaterial vom Dokumentenmanagement komplett getrennt abläuft.
- **Informelles vs. formelles Lernen.** Auch die pädagogischen Ansätze zur Unterstützung von formellem und informellem Lernen passen oft nicht zusammen, und auch die Ansätze kollaborativer Lernlösungen unterscheiden sich erheblich von Kollaborationswerkzeugen aus dem Wissensmanagement.
- **Wissensmanagement vs. Personalentwicklung.** Obwohl Skills-Management oft als eine Unterdisziplin des Wissensmanagements aufgefasst wird, unterscheidet sich beispielsweise das Verständnis von Kompetenzen bei beiden erheblich. Während im eher von der Personalentwicklung bestimmten Skills-Management langfristige Kompetenzen im Vordergrund stehen, ist im Bereich des Ad-Hoc-Lernens und der frühen Phasen des Prozesses eine feingranularere Sichtweise vonnöten. Eine Verbindung der beiden findet oft nicht statt, so dass in informellen Lernprozessen erworbene Kompetenzen nicht dokumentiert oder honoriert werden.

Will man diese Brüche zumindest ansatzweise überwinden, so ist die Lösung auf mehreren Ebenen zu suchen. Auf der *konzeptuellen Ebene* ist ein Modell zu entwickeln, so dass sich die einzelnen Disziplinen als komplementär verstehen und an den Schnittstellen auch miteinander »reden« können. Hier bietet der Reifungsprozess eine Orientierung, wo diese Schnittstellen zu finden sind und was sich zwischen den einzelnen Phasen ändert. In Abb. 3.10 ordnet der Wissensreifungsprozess die unterschiedlichen Disziplinen, die sich mit dem Lernen in Organisationen beschäftigen so an, dass sichtbar wird, wo der jeweilige Schwerpunkt zu finden ist. Während sich Wissensmanagement und Organisationales Lernen auf die frühen Phasen und informellen Lernformen konzentrieren, ist der Schwerpunkt von traditionellem E-Learning eher im formellen Bereich gegeben, wo sich auch die Personalentwicklung befindet. Diese Zusammenschau der unterschiedlichen Disziplinen ist der erste Schritt zur Überwindung der Brüche, die durch die jeweils unterschiedlichen Konzeptionen begründet werden werden.

Aber auch auf *technischen Ebene* ist eine reifungsbewusste Lernunterstützung notwendig, die Barrieren, die sich z.B. aus der mangelnden Verfügbarkeit ergeben, überwinden helfen. Wie das Modell der einzelnen Reifungsschritte gezeigt hat, ist für den Reifungsvorgang die Situation zentral. Die Situation bestimmt, welche Artefakte für eine Reifung erforderliche

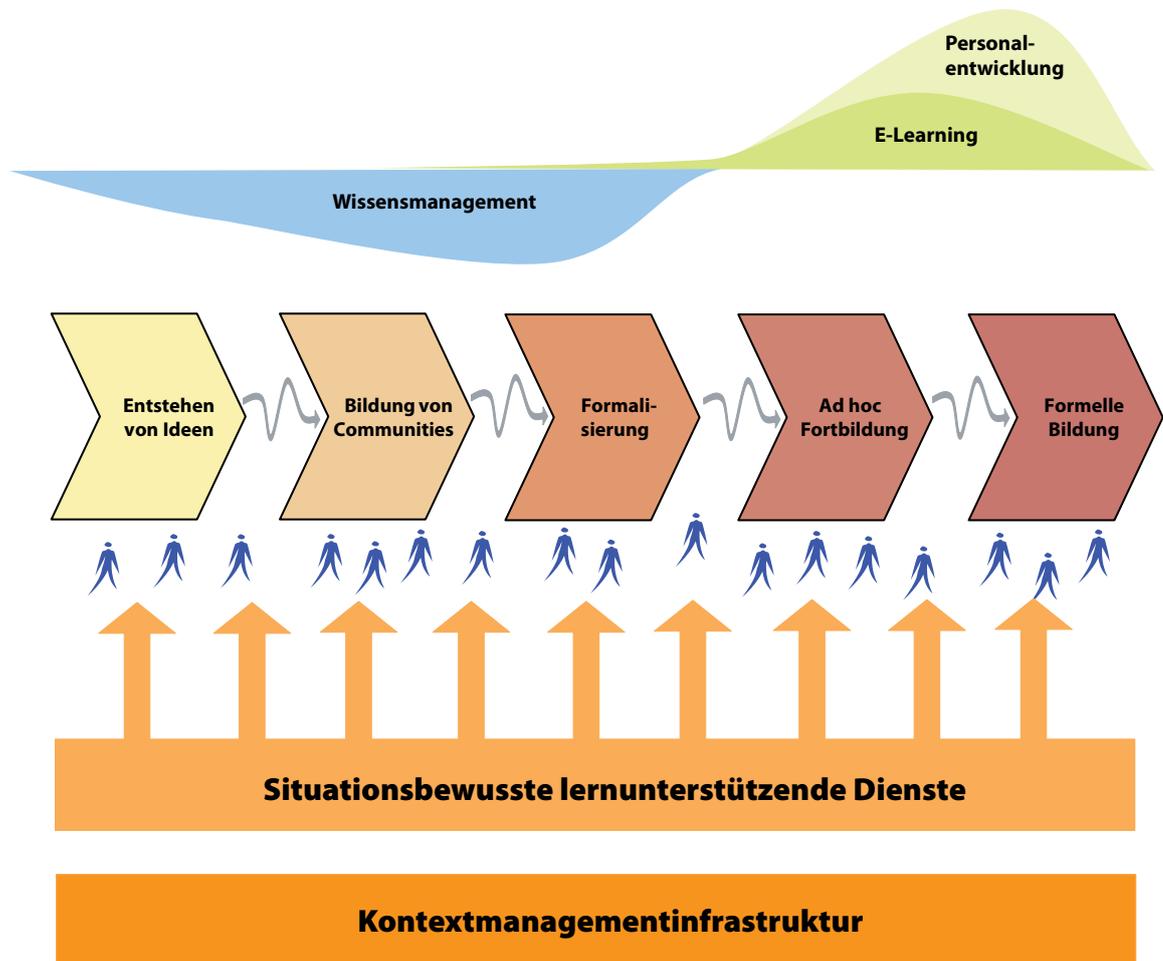


Abbildung 3.10.: Situationsbewusste Unterstützung des Wissensreifungsprozesses

Bausteine sind. Dabei sind Artefakte desto mehr mit ihrer Entstehungssituation verhaftet, je »unreifer« sie sind. Der Schlüssel für eine lernunterstützende technische Infrastruktur ist also die Operationalisierung dieser Situation (was im folgenden als »Kontext«⁴ bezeichnet werden soll) und deren Bereitstellung für unterschiedliche darauf aufsetzende Dienste.

Der so durch eine sog. »Kontextmanagementinfrastruktur« bereitgestellte Kontext kann durch die Dienste an unterschiedlichen Stellen eingesetzt werden:

- Er bildet die Grundlage für die Selektion (was ist relevant) und ggf. Anpassung der

⁴Eine präzisere Definition wird im Rahmen einer genaueren Analyse der Abbildungsproblematik im weiteren Verlauf der Arbeit noch gegeben.

Ressourcen an die Situation (wie kann man das Relevante optimal präsentieren). Hierbei kann Selektion und Präsentation sowohl Informationsressourcen als auch menschliche Ressourcen umfassen. Ein Beispiel ist hierfür die Auswahl von Lernressourcen, die genau das vermitteln, was in der aktuellen Situation vom Lernenden aufgrund seiner Vorkenntnisse und der Wissensanforderungen gefordert wird. Die Anpassung kann hier das Ergänzen um diejenigen Materialien mit einschließen, die das notwendige Verständnis für die Lernressourcen (Voraussetzungen) vermitteln, so dass der Benutzer nicht durch für ihn Unverständliches überfordert wird.

- Ebenso ist der Benutzerkontext für die Ermittlung des Erstellungskontextes essentiell. Dieser Erstellungskontext annotiert Informationsressourcen, so dass bei der Selektion darauf zurückgegriffen werden kann. Grundlage hierfür ist die Annahme, dass Erstellungskontext und Verwendungskontext einander ähneln bzw. dass die Ähnlichkeit die Relevanz bestimmt. Dies trifft beispielsweise auf prozessbezogene Informationen zu.
- Wenn man Lernen makroskopisch als eine Abfolge von Nutzung und Erstellung von Ressourcen ansieht, kann man durch die Erfassung des Erstellungskontextes auch »Ketten« bilden: man kann nachverfolgen, woher bestimmte Inhalte kommen, auch ohne dass das explizit angegeben wäre.

Im folgenden Verlauf der Arbeit soll dieses Potential anhand eines Ausschnitts aus dem Wissensreifungsprozess illustriert werden.

3.7. Fazit

Durch die Analyse des Wissensreifungsprozesses wurde die Rolle von Information in Form von Artefakten noch weiter in den Vordergrund gerückt. Sie können die Gestalt von informellen »Schnippseln« oder pädagogisch aufbereiteten Lernobjekten haben, also einen unterschiedlichen Reifegrad als Abbild des Wissenreifegrades im Ersteller. Typischerweise ist das Reifen auch kein streng linearer Prozess, der sich von Artefakt zu Artefakt entlanghangelt. Vielmehr ist es eingebettet in Problemlösungsprozesse, für die die Identifikation relevanter Artefakte nicht-trivial ist. All dies macht eine genauere Analyse von Informationssuchprozessen und ihre Beziehung zu Lernprozessen notwendig. Zudem existiert hier ein recht großer Erfahrungsschatz, den es einzubringen gilt.

Zuvor soll allerdings im folgenden Kapitel genauer herausgearbeitet werden, auf welchen Teil des Wissensreifungsprozesses sich der Rest der Arbeit konzentriert, um zu zeigen, dass und wie situationsbewusste Systemdienste Lernprozesse unterstützen.

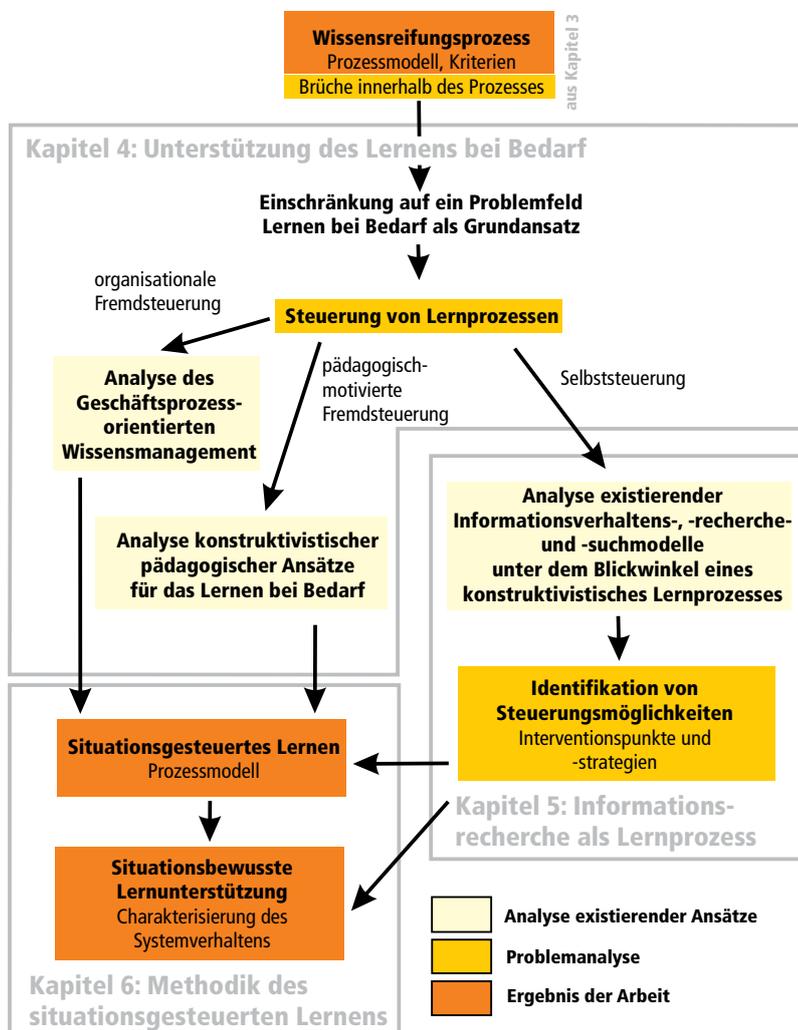
Teil II.

Situationsgesteuertes Lernen: Eine Methode für die Unterstützung des Lernens bei Bedarf

Problemstellung

These ist: Brüche im Wissensreifungsprozess lassen sich durch eine geeignete Lernunterstützung vermeiden/überwinden. Hierfür wird eine geeignete Methodik benötigt, die exemplarisch für eine Form des arbeitsbegleitenden Lernens (das Lernen bei Bedarf) erarbeitet werden muss. Die skizzierte Rolle der jeweiligen Lern-/Arbeitssituation muss hierzu näher analysiert werden. Diese Methodik muss weiterhin klare Ansatzpunkte für eine informationstechnische Umsetzung liefern.

Vorgehensweise und Struktur



4.

Unterstützung des Lernens bei Bedarf

Der Wissensreifungsprozess bietet eine Orientierung für die Vielfalt des Lernens in betrieblichen Umgebungen. Es wäre vermessen, eine Technologieunterstützung für den gesamten Prozess anzustreben; deshalb wollen wir an dieser Stelle das »Lernen bei Bedarf« herausgreifen, das bereits in den ersten beiden Kapiteln als ein angemessenes Paradigma für die aktuellen Herausforderungen an die betriebliche Personalentwicklung erwähnt wurde. Es findet sich an der Schnittstelle von klassisch verstandenem E-Learning und Wissensmanagement, zwischen Formalisierungs- und Ad-hoc-Fortbildungsphase. Hierzu soll zunächst präziser gefasst werden, was darunter zu verstehen ist, bevor wir anhand des Modells des Wissensreifungsprozesses herausarbeiten, welche Ansatzpunkte für eine situationsbewusste Unterstützung des Lernens bei Bedarf existieren und daraus eine Methodik ableiten: das »situationsgesteuerte Lernen«.

4.1. Lernen bei Bedarf und die Steuerung von Lernprozessen

4.1.1. Begriff des Lernens bei Bedarf

Was heißt denn nun eigentlich »Lernen bei Bedarf«? Beim Lernen bei Bedarf wird Wissen erst dann durch den jeweiligen Lernenden aufgebaut, wenn es auch zur Anwendung benötigt wird. Dies ist motiviert durch die zunehmende Kritik an einem Lernen auf Vorrat in großen quasi-standardisierten Einheiten (Kurse, Seminare), da es zum einen zu einer Trennung von Arbeiten und Lernen, zum anderen oft zu sog. »trägem Wissen« führt, wie es für dekontextualisierte Lernumgebungen typisch ist, da das Wissen eben nicht in realen Situationen aufgebaut wird, in denen es auch angewandt werden soll [Ren96]. Im einfachsten Fall heißt dies, dass anstatt einer gesteuerten Weiterbildung man es dem Mitarbeiter überläßt, lernunterstützende Maßnahmen nachzufragen: der Mitarbeiter versucht, sobald er eine Wissenslücke wahrnimmt, nach entsprechenden Hilfen zu suchen. Hierbei bemüht

man bewusst die Informationsrecherche als Analogie. Deshalb wird in einem ersten Schritt (im folgenden Kapitel) diese analysiert, und zwar unter der Grundannahme, dass die Informationsrecherche als ein Lernprozess aufgefasst werden kann und muss.

Es zeigt sich, dass wenn man Informationsrecherche als ganzheitlichen Lernprozess auffasst, man zu einer ähnlichen Schlussfolgerung kommt, die der Pädagogik zugrundeliegt: Lernen kann durch gezielte Führung gefördert und verbessert werden, weshalb eine Methode entwickelt und präsentiert wird (die als »situationsgesteuertes Lernen« bezeichnet wird), die eine situationsbewusste Führung durch das lernunterstützende Rechnersystem erlaubt. Diese positioniert sich bewusst zwischen der traditionellen Vorstellung vom Kurslernen (als komplette Kurssteuerung) und der naiv verstandenen Informationsrecherche (als komplette Selbststeuerung).

4.1.2. Steuerung von Lernprozessen

Um dies präziser zu erläutern, soll kurz auf das Wesen der Steuerung von Lernprozessen eingegangen werden. Aufbauend auf den »Dimensionen der Selbststeuerung« von KRAFT [Kra99] lassen sich die Steuerungsaufgaben innerhalb von Lernprozessen wie folgt klassifizieren:

- **Lernzielbestimmung und Inhaltsauswahl.** Es wird festgelegt, welchen Zweck Lernprozesse verfolgen und was bei ihrem erfolgreichen Verlauf als Ziel erreicht werden soll. Weiterhin umfasst dies auch die Auswahl geeigneter Ressourcen für diese Lernziele.
- **Lernkoordination.** Dies umfasst die Abstimmung der Lernaktivitäten mit anderen Aktivitäten, also in dem hier betrachteten Szenario besonders das Zusammenwirken von Lernen und Arbeiten.
- **Lernorganisation.** Dies bezeichnet die Mikrosteuerung der einzelnen Schritte im Lernprozess. Im Falle der Selbststeuerung trifft der Lernende Entscheidungen über Lernort, -zeitpunkt, -tempo und evtl. Lernpartner.
- **Lernerfolgskontrolle.** Anhand der Lernziele wird überprüft, ob der Lernprozess zum gewünschten Erfolg geführt hat.
- **Subjektive Interpretation der Lernsituation.** Hierbei geht es um die subjektive Wahrnehmung durch den Lernenden, ob er sich als der Gestalter seiner Lernprozesse sieht oder diese als fremdgesteuert wahrnimmt.

Die tatsächliche Steuerung kann dabei sowohl personell als auch strukturell erfolgen. *Personelle Steuerung* liegt dann vor, wenn – wie im klassischen Lehrer-Schüler-Modell –

ein pädagogischen Rollenträger ein oder mehrere Steuerungsaufgaben wahrnimmt. Dies kann durch die Vermittlung in Präsenzveranstaltungen geschehen, aber auch durch die Zuweisung von durchzuarbeitenden Lernressourcen. *Strukturelle Steuerung* liegt dann vor, wenn die vorgegebenen Strukturen der Lernressourcen wie auch der Möglichkeiten zur Lernorganisation (bestimmt durch technische und organisatorische Rahmenbedingungen) maßgeblichen Einfluss auf den Verlauf von Lernprozessen nehmen. Typisches Beispiel sind hier Kursstrukturen in Form einer Abfolge der einzelnen Module/Lektionen oder aber die Organisationsformen, die von Lernmanagementsystemen angeboten werden.

Nach einer Untersuchung derzeitiger E-Learning-Konzeptionen im betrieblichen Lernen lassen sich folgende Typen von Steuerungskonzepten identifizieren:

- **Kursgesteuertes Lernen.** Dies ist sicherlich derzeit die am weitesten verbreitete Form des Lernens. Die Lernaktivität wird durch die vorgegebene Kursstruktur bestimmt. Kurse sind dabei i.d.R. relativ lange Lerneinheiten, die abonniert oder zugewiesen werden. Dies kann sich sowohl auf Präsenzkurse als auch elektronische Kurse (oder »Blended Learning«-Lösungen) beziehen.
- **Selbstgesteuertes Lernen.** Hierbei sucht der Lernende selbst aktiv nach Lerneinheiten, die sein momentanes Wissensbedürfnis zu befriedigen helfen. Er bestimmt vollständig Lernziele, -zeitpunkt, -ort und führt auch eine evtl. Lernkontrolle eigenständig durch (vgl. auch [Erp97]).
- **Ungesteuertes Lernen.** Diese Art der Lernform ist subtiler und weniger bewusst als die anderen Formen. Ungesteuertes Lernen findet vor allem in sozialen Prozessen statt, so z.B. als eine unverbindliche Unterhaltung beim Essen oder in der Kaffeepause. Hier wird vor allem informelles, nicht explizites Wissen ausgetauscht. Es existieren keine Lernziele und keine Erfolgskontrolle.

Daran erkennt man, dass derzeit im wesentlichen die beiden Pole »Kurssteuerung« und »Selbststeuerung« existieren, wobei derzeit die Entwicklung klar in Richtung Selbststeuerung geht. Allerdings sind beide Extreme nicht optimal. Im Falle des kursgesteuerten Lernens erfolgt die Strukturierung (und die daraus resultierende Steuerung) ohne Berücksichtigung der konkreten Arbeitssituation und zudem meist in relativ großen Einheiten. Dies eignet sich nicht für die Integration mit den Arbeitsprozessen, sondern führt praktisch immer zu einer Trennung der beiden Aktivitäten – und sei es nur in Form von »Lernzeiten« am Arbeitsplatz. Umgekehrt ist selbstgesteuertes Lernen in seiner Extremform sehr anspruchsvoll: es erfordert eine hohe Lern- und Selbstorganisationskompetenz. Weiterhin lassen sich hier die Ergebnisse aus der Informationsverhaltensforschung übertragen, d.h. Lernprozesse werden durch die kognitiven wie affektiven Barrieren abgebrochen oder gar nicht erst initiiert. Auch wenn KRAFT betont, dass »selbstgesteuertes Lernen [...] ja gerade

nicht [heißt], Lernende allein zu lassen im Lernprozess« [Kra99], so legt die Begriffsbildung und die Praxis dieses Missverständnis eben doch nahe.

4.1.3. Fokus der Arbeit

Deshalb ist das Ziel im weiteren Verlauf der Arbeit die Vorstellung einer Methode, die sich bewusst zwischen den beiden Extremen positioniert und die eine Verzahnung von Lernen und Arbeit ermöglicht, ohne dass der einzelne überfordert wird: das sog. **situationsgesteuerte Lernen**. Grundidee hierbei ist, dass das System die Arbeit des Mitarbeiters verfolgt und den jeweiligen Wissensstand vergleicht mit den Wissensanforderungen. Entdeckt das System eine Wissenslücke, so empfiehlt es geeignete **Lerngelegenheiten**, wobei der Begriff der »Lerngelegenheit« als Oberbegriff verstanden wird für explizite Lernressourcen (wie klassische Lernobjekte, Lernprogramme im Sinne des E-Learning [Kno04], didaktisch nicht oder wenig aufbereitete Dokumente, Objekte der realen Welt wie Bücher, aber auch Präsenzseminare) und Personen (in Form von zugewiesenen Tutoren oder als Experten). Dementsprechend impliziert Lernunterstützung im Sinne dieser Arbeit nicht, dass Lernprozesse ausschließlich am Rechner stattfinden müssen, sondern nur durch das Rechnersystem beeinflusst werden. Der Mitarbeiter kann dann selbst entscheiden, ob er der Empfehlung folgt und in einen Lernprozess einsteigt.

Obwohl Lernen bei Bedarf sicherlich in allen Phasen des Wissensreifungsprozesses eine Rolle spielt, ist die Methodik des situationsgesteuerten Lernens am vielversprechendsten für die Formalisierungs- und Ad-Hoc-Phase. Durch den erreichten Reifegrad des Wissens, das in diesen Phasen erworben und weitergegeben wird, ist das Verständnis auch soweit fortgeschritten, dass man Teilen der Arbeitssituation Wissensanforderungen zuordnen kann. So kann das System in die Lage versetzt werden, einzuschätzen, was in der Situation erforderlich ist.

4.2. Szenarien

Dies soll im folgenden an drei Szenarien illustriert werden, die im weiteren Verlauf der Arbeit als Beispiele herangezogen werden. Diese basieren im wesentlichen auf den Szenarien, die im Rahmen des Projektes *Learning in Process* für die Erprobung und Evaluierung herangezogen wurden, sowie der Fallstudie bei einem Logistikdienstleister [Win02]:

- **Szenario 1** zeigt, wie sich klassische Personalentwicklungskonzepte durch ein Lernen bei Bedarf flexibler und dynamischer gestalten lässt (z.B. zum Umgang mit erhöhter Mitarbeiterfluktuation), ohne dass sich die Komplexität der Prozesse erhöhen muss.

- **Szenario 2** zeigt, wie man prozessorientiert bei Bedarf Mitarbeiter qualifizieren lassen kann, was vor allem für neue Mitarbeiter oder Urlaubsvertretungen relevant ist.
- **Szenario 3** zeigt, wie auch in weniger strukturierten Umgebungen Aufgaben und ihre Anforderungen erkannt werden können.

4.2.1. Szenario 1 – Kundenberater im Logistikvertrieb: Bedarfsorientierte Empfehlungen auf der Basis von Rollen

In traditionellen Konzeptionen von Personalentwicklung wird meist zentral (von Vorgesetzten oder Personalentwicklungsverantwortlichen) vorgegeben, wer wann welche Fortbildung wahrzunehmen hat. So werden für neue Mitarbeiter oder Wechsler aus anderen Abteilungen Entwicklungspläne erarbeitet, die die Einzelschritte festlegen. Auch sind Qualifizierungsmaßnahmen meist integraler Bestandteil von Projektteamzusammenstellungen. Dies funktioniert auch relativ gut, solange die Umgebung stabil ist, d.h. nicht zu oft wechselnde Projekte, nicht zu schnell wechselnde Anforderungen und nicht zu schnell wechselnde Mitarbeiter; ansonsten lässt sich das nicht mehr so einfach steuern.

Mit einem Ansatz, der dem »Lernen bei Bedarf« als Grundparadigma folgt, könnte man sich das auch anders vorstellen. Betrachten wir z.B. Firmen mit hoher Personalfuktuation, in denen beispielsweise viel mit Teilzeit- und Aushilfskräften (z.B. Werkstudenten) gearbeitet wird. Ein Beispiel sind Logistik-Dienstleister im Bereich der Kundenberatung, wie in [Win02] analysiert wurde. Die dort angebotenen Produkte sind zum einen häufigeren Veränderungen in ihren Bedingungen und ihrer Preisgestaltung unterworfen, zum anderen gerade im internationalen Bereich wissens- und beratungsintensiv. So existieren beispielsweise saisonale Produkte z.B. für den Grußkartenversand, für den besondere Bedingungen beim Versand in islamische Länder gelten (z.B. bzgl. Motive). Die Kundenberatung ist durch flexible Zuordnung von zuständigen Mitarbeitern gekennzeichnet, die zwar durchaus ihre Spezialgebiete haben, allerdings häufig auch bei anderen Gebieten einspringen müssen. In einer solchen Umgebung sind traditionelle Weiterbildungskonzepte entweder zu aufwendig oder einfach nicht flexibel genug, um sicherzustellen, dass die Mitarbeiter auch über die Kompetenzen¹ verfügen, die sie für eine qualifizierte Beratung benötigen.

Lernen bei Bedarf könnte sich hier wie folgt vollziehen:

- Der Wissensbedarf bzw. die Kompetenzanforderungen werden für die einzelnen Produkte bzw. Produkttypen spezifiziert, wobei durchaus (wie z.B. bei kulturell bedingten Einschränkungen) nicht jede Anforderung produktspezifisch sein muss.

¹Wie in Kap. 10 näher ausgeführt wird, soll hierbei unter »Kompetenzen« nicht eine kleine Menge von generischen Kompetenzen im Sinne von Schlüsselkompetenzen bzw. -qualifikationen [Mer74] verstanden werden, sondern feingranularere Beschreibungen der Verhaltensdispositionen von Mitarbeitern.

- Mitarbeitern werden die Zuständigkeiten für diese Produkte im Rahmen der Einsatzplanung zugewiesen.
- Bei den Mitarbeitern wird abgeglichen, über welches Wissen bzw. welche Kompetenzen sie bereits verfügen – und ihnen werden empfohlene Lernobjekte für die fehlenden Kompetenzen angezeigt. Diese können sofort bearbeitet werden bzw. auf eine Phase verschoben werden, in der die Arbeitsbelastung geringer ist.
- Ähnlich können auch Kompetenzen aktualisiert werden, wenn beispielsweise sich nur geringfügige Aspekte für ein Produkt geändert haben. In diesem Fall kann man einem Mitarbeiter nur ein Aktualisierungsobjekt bereitstellen, das weniger umfangreich ist und auf die Änderungen zugeschnitten ist (z.B. neues Preismodell).

Mit diesem Mechanismus können auch Phänomene erfasst werden, dass Kompetenzen, die über längere Zeiträume nicht aktiviert wurden, auch verloren gehen können, so dass eine erneute Schulung erforderlich ist.

4.2.2. Szenario 2 – Projektmanagement: Prozessorientiertes Lernen bei Bedarf durch Erkennen des Prozesskontextes

Während im vorangegangenen Szenario vieles manuell zu erfolgen hatte, wie z.B. die Zuweisung von Zuständigkeiten, und auch sehr stark auf eine asymmetrische Rollenverteilung zwischen den von der Organisation »verordneten« Inhalten und den »konsumierenden« Mitarbeitern fokussiert war, eröffnet die Betrachtung von Geschäftsprozessen und die Erkennung der Prozessschritte des jeweiligen Mitarbeiters erweiterte Möglichkeiten.

Ein Beispiel sind hier Portale für das Projektmanagement, in denen unterschiedliche Aufgaben prozessorientiert angeboten werden, angefangen mit Anforderungsermittlung und -verwaltung über das Berichtswesen (z.B. hinsichtlich Ressourcenverbrauch), was je nach Auftraggeber bzw. Projektträger unterschiedlichen Anforderungen zu genügen hat, bis hin zum Release-Management und Bugtracking. Da diese einzelnen Prozesse und Prozessschritte ohnehin in der Portalanwendung modelliert sind, kann man sie als Kontext des Benutzers abgreifen, sobald der Benutzer sich in den entsprechenden Portalbereich begibt. So kann beispielsweise einem Mitarbeiter beim Eintritt in die Anforderungsanalyse ein Lernobjekt oder eine bald stattfindende Schulung zu diesem Thema angeboten werden. Sind Berichte im Rahmen von EU-Forschungsprojekten zu erstellen, so kann das System andere Mitarbeiter empfehlen, die kürzlich dasselbe getan haben oder sich schon länger damit auskennen. Auch können Vorlagen, Musterdokumente oder Beispiele aus früheren Projekten empfohlen werden, um beispielsweise die Organisation eines Projekttreffens zu erleichtern. Die Empfehlungen können dabei direkt in der Portalanwendung angezeigt werden.

Hierbei ist zu beachten:

- Die jeweiligen Empfehlungen sind darauf abgestimmt, was der Mitarbeiter bereits kann, da es keinen Sinn macht, jemanden einen Grundlagenkurs zur Anforderungsanalyse zu geben, der dies schon in etlichen Projekten (erfolgreich) getan hat.
- Umgekehrt allerdings machen Dokumente auf »niedrigeren« Reifestufen wie Beispieldokumente oder Vorlagen erst für erfahrenere Mitarbeiter Sinn, die über das nötige Grundlagenwissen verfügen.
- Ebenso können Spezialthemen (wie z.B. Berichte für EU-Projekte) erst dann vermittelt werden, wenn die Grundlagen zu Ressourcenplanung bei Projekten verstanden ist.
- Beim Empfehlen anderer Personen ist darauf zu achten, dass diese auch verfügbar sind, da ansonsten gerade akute Probleme nicht gelöst werden können.

4.2.3. Szenario 3 – Softwareentwicklung: Erkennen von aktuellen Aufgaben

Viele Arbeitsprozesse lassen sich nicht rein prozessorientiert charakterisieren, da ihr Strukturierungsgrad deutlich geringer ist und durch häufige Wechsel zwischen unterschiedlichen Aktivitäten gekennzeichnet sind. Dies gilt sicherlich z.B. auch für Softwareentwicklungstätigkeiten, soweit sie über die durch Projektmanagement definierten grob granularen Prozesse hinausgehen. Hier tritt oft das Problem auf, dass mit Technologien umgegangen werden muss (z.B. zur Wartung von existierendem Code, zur Nutzung von existierenden Komponenten o.ä.), mit denen der jeweilige Entwickler nur begrenzte Erfahrungen hat. Hier kann das System aus dem aktuellen Quelltext und Aktionen des Benutzers erkennen, was er denn gerade tun will, um daraus den Kompetenzbedarf zu ermitteln. Auch kann der Kontext im aktuellen Verzeichnisbaum (z.B. bei Entwurfsdokumenten) ein Anhaltspunkt für die aktuelle Aufgabe sein.

Hier ist zu beachten:

- Die Methoden, um herauszufinden, was der Benutzer gerade tut, sind heuristischer Natur und damit nicht hundertprozentig zuverlässig.
- Spezialthemen (wie z.B. Kenntnisse über eine bestimmte Ontologie-API) können erst dann vermittelt werden, wenn die Grundlagen zu Ontologien bzw. einer bestimmten Ontologiesprache verstanden sind.

4.3. Existierende methodische Ansätze für das Lernen bei Bedarf

Wie man an den Szenarien erkennen konnte, verschwinden hier die Grenzen zwischen E-Learning und Wissensmanagement zunehmend, da das situationsgesteuerte Lernen sich sowohl für die Empfehlung von Ansprechpartnern (klassische »Expert-Finder«), von Gebrauchsdokumenten (klassisches dokumentenorientiertes Wissensmanagement) als auch von didaktisch gestalteten Lernobjekten und Teilen von Kursen (klassische E-Learning-Domäne) eignet. Dies ist auch sinnvoll, da gerade der Übergang zwischen Formalisierungs- und Ad-Hoc-Phase durch die Trennung in unterschiedliche Disziplinen durch Barrieren erschwert war.

Aus dieser integrativen Sichtweise sollen im folgenden auch vor der Entwicklung der Methodik zunächst im folgenden Abschnitt didaktische Ansätze sowie Ansätze aus dem Wissensmanagement (als Vertreter einer pädagogisch bzw. organisational motivierten Fremdsteuerung) analysiert werden, bevor im folgenden Kapitel die Informationsrecherche (als Lernprozess interpretiert) als Vertreter eher selbstgesteuerter Ansätze auf ihre Beiträge hin untersucht wird.

4.3.1. E-Learning-Ansätze: pädagogisch motivierte Fremdsteuerung

Situiertes Lernen

Das Konzept des »situierten Lernens« geht von einem konstruktivistischen Grundverständnis des Lernens aus. Das grundlegende Postulat hierbei ist, dass Lernen nicht ablösbar von der Aktivität, der Situation und der Kultur, innerhalb dessen es sich vollzieht, verstanden werden kann [LW91]. Dies wird mit dem Begriff der »Situiertheit« beschrieben. Für die Konzeption von didaktischen Szenarien ergibt sich daraus die Folgerung, dass sich Wissen, Wissenserwerb und Anwendung nicht voneinander trennen lassen [Blu98] und für Wissenserwerb und Anwendung idealerweise dieselbe bzw. eine ähnliche Situation als Hintergrund dienen sollte [DJ92], so dass die reale Situation das Lernen motiviert und sich der Lernerfolg in der unmittelbaren Anwendung auf die reale Problemsituation zeigt.

Speziell der Ansatz der sog. »anchored instruction« versucht, das Prinzip des situierten Lernens auf schulische/akademische Umgebungen zu übertragen. Kernziel ist hierbei die Vermeidung von trägem Wissen, das zwar erworben wurde, aber nicht angewandt und für reale Problemlösungsprozesse eingesetzt werden kann. Lerninhalte sollen daher in sinnvollen, problemorientierten und lebensnahen Kontexten verankert werden, so dass die Lernenden motiviert werden, eigenständig Themen zu explorieren, Probleme zu konstruieren und Lernergebnisse zu präsentieren [Röl03].

Als pädagogische Fundierung bildet das Konzept des »situierten Lernens« einen idealen

Ausgangspunkt für das arbeitsbegleitende Lernen. Gerade die Arbeitsprozesse bieten sich als natürliche »Problemsituationen« an, an denen sich das Lernen orientiert. Anders als in schulischen und akademischen Umgebungen müssen so keine künstlichen »realitätsnahen« Situation geschaffen werden, sondern sie liegen unmittelbar vor und können – wenn man sie nur hinreichend einbeziehen kann – als Ankerpunkt des Lernens genutzt werden.

Scaffolding

Scaffolding [McK99] (vgl. Kapitel 2) ist ein Ansatz aus dem Umfeld konstruktivistischer Lernumgebungen und bezeichnet Lehrtätigkeiten, die versuchen, Lernenden in offenen Umgebungen Strukturen zu geben, an denen sie sich entlanghangeln können, um so die Orientierungslosigkeit beim selbstgesteuerten Lernen zu überwinden. Scaffolding kann darin bestehen, dass dem Lernenden klar Sinn und Zweck von Lernaktivitäten erläutert werden, ihm Referenzpunkte und -richtungen o.ä. vorgegeben werden. Mit dieser Grundkonzeption ist Scaffolding ein ideales pädagogisches Paradigma für die Lernunterstützung in offenen Umgebungen, so dass es in dieser Arbeit im wesentlichen um eine mögliche Umsetzung dieses Paradigmas auf Scaffolding durch Rechnersysteme geht.

Als einen ersten Ansatz für ein automatisiertes Scaffolding könnten **pädagogische Agenten** angesehen werden. »Pädagogische Agenten sind Softwareagenten, die Wissen vermitteln und Benutzer im Lernbereich anleiten und begleiten.« [Ben04]. Die Grundkonzeption ist dabei, dass dem Benutzer eine autonome Softwarekomponente zugeordnet wird, die den Lernprozess unterstützen soll. Diese Unterstützung kann dabei sowohl reaktiv als auch proaktiv unter Zuhilfenahme externer Ressourcen oder anderer Agenten [Ben03] erfolgen. Die Aktionsmöglichkeiten solcher Agenten werden wie folgt klassifiziert: [Ben04]

- **Information.** Der pädagogische Agent sammelt, strukturiert und vermittelt Informationen, die im Lernzusammenhang relevant sind und der Befriedigung von Wissens- und Informationsbedürfnissen des Benutzers bzw. der Bewältigung von Problemen und der Herstellung von Handlungskompetenz dienen.
- **Kommunikation.** Pädagogische Agenten kommunizieren mit dem Anwender bzw. mit anderen Agenten und Systemen, wobei je nach Kommunikationspartner inhaltliche, fachliche, persönliche, strukturelle, methodische oder systembezogene Fragen eine Rolle spielen.
- **Transaktion.** Die pädagogischen Agenten bereiten im Lernbereich Transaktionen vor und führen sie durch; es werden im Auftrag des Lernenden Content, Services und Produkte gesucht und vermittelt sowie Ressourcen zwischen Benutzern ausgetauscht.

- **Interaktion.** Der pädagogische Agent agiert in Wechselwirkung mit Benutzern, Agenten und Systemen, um Zustandsveränderungen und Analysen zu erreichen, die dem Benutzer beim Lernen und Arbeiten unmittelbar oder mittelbar dienlich sind.

Der Anwendungsbereich wird neben der tutoriellen Assistenz innerhalb von Lernprogrammen bzw. Lernplattformen von BENDEL im sog. »Just-in-Time-Learning« gesehen [Ben02], wo insbesondere die Funktionen Information und Kommunikation als wichtig identifiziert werden.

Die Forschung zu pädagogischen Agenten konzentriert sich stark auf anthropomorphe Darstellung und Animation (sog. »lifelike agents«) und die hierüber kommunizierbaren affektiven Inhalte. Die Frage, wie das am Benutzer orientierte Verhalten realisiert wird, wie also z.B. für die Situation relevante Inhalte gefunden werden, wie mögliche Ansprechpartner etc. lokalisiert werden, bleibt allerdings offen.

Das Konzept der pädagogischen Agenten ist ein sehr guter Ausgangspunkt für eine begleitende Lernunterstützung, die unterschiedliche Grade der Führung realisieren lässt. Ob der hohe Grad der Autonomie solcher pädagogischer Agenten unbedingt notwendig ist, ist fraglich. Gleiches gilt für eine anthropomorphe Darstellung. Umgekehrt muss viel stärker untersucht werden, wie die Beziehung zwischen den zu initiiierenden Lernprozessen und der Arbeitssituation als Ausgangspunkt aussieht.

Microlearning

Der Trend in Richtung zu kleinen und modularen Lerneinheiten wird seit kurzem auch mit dem Begriff »Microlearning« umschrieben [Hug05], wobei die Begriffsbildung noch sehr unscharf ist und von didaktisch aufbereitetem Lernen in stationären oder mobilen Kontexten (z.B. via Mobiltelefonen) bis hin zu selbstgesteuertem Lernen in seiner Extremform reicht. Als »Mikroinhalte« kommt dabei auch das gesamte Spektrum, das von kleinen, eher klassisch entworfenen Lernobjekten bis hin zu typischen Inhalten, wie sie in »Web 2.0«-Anwendungen wie Weblogs oder Wikis entstehen, oder anderen »Informationsschnipseln« reicht.

Ingesamt ist beim »Microlearning« noch von keinem konkreten Ansatz auszugehen (und noch weniger von einem konkret fassbaren technischen Ansatz), sondern eher von einem Überbegriff für ein emergentes Phänomen, für das auch diese Arbeit ein Stellvertreter ist: der Trend weg von kursorientierten, großen und standardisierten Lerneinheiten hin zu kleinen, flexibel und situationsbewusst einsetzbaren Lerninhalten.

4.3.2. Geschäftsprozessorientiertes Wissensmanagement: organisational motivierte Fremsteuerung

Mit der generellen Hinwendung zu den Geschäftsprozessen als Erfolgsfaktoren hat sich Ende der 90er Jahre das »geschäftsprozessorientierte Wissensmanagement« herausgebildet, das eine Methodik und eine Familie von technischen Ansätzen beschreibt, die das Wissensmanagement zur Unterstützung bei der Ausführung von Prozessen mit dem Geschäftsprozessmanagement als Management des Wissens über Prozesse verbinden will ([AHMM02], [Rem02], [Abe04]). Auf methodischer Ebene geht es dabei um die Frage, wie sich ein Zusammenhang zwischen Wissensartefakten und Geschäftsprozesselementen herstellen und somit eine strukturierte Analyse als Voraussetzung für ein erfolgreiches Wissensmanagement realisieren läßt. So stehen auch Prozessanalyse und -modellierung im Zentrum, auf deren Basis gezielte Interventionen bestimmt werden. Auf technischer Ebene werden geschäftsprozessorientierte Informationszugänge bereitgestellt, z.B. in Form von Prozessassistenten. Hierbei werden meistens explizite Prozessmodelle und Workflow-Systeme zu deren Ausführung vorausgesetzt. Neben den Prozessen als Strukturierungsinstrument ist vor allem das organisationale Gedächtnis (*organizational memory*) ein zentrales Konzept. Dieses soll mit Hilfe der Prozesse so strukturiert werden, dass ein effizienter Zugriff aus operativen Prozessen heraus möglich ist. Primär sind die technischen Systeme damit also sog. »OMIS« (*Organizational Memory Information Systems*).

Für den Endbenutzer zeigen sich diese Systeme dadurch, dass sie (gesteuert durch Informationen auf dem Workflow-System) dem Benutzer Dokumente empfehlen, die zum aktuellen Prozessschritt passen. Bei diesen Dokumenten kann es sich um Best-Practices oder andere Erfahrungsdokumentationen handeln oder auch um verwandte Fälle, die als Anschauungsmaterial dienen können.

Wichtige Repräsentanten dieser Ansätze sind:

- Wegweisend für die Methode des geschäftsprozessorientierten Wissensmanagement waren Forschungsprojekte, die am DFKI durchgeführt wurden und die im übernächsten Kapitel auf technischer Ebene noch näher analysiert werden: **KnowMore** (Knowledge Management for Learning Organizations, vgl. z.B. [ABS99]) fokussiert auf die aktive Informationsbereitstellung durch ein System bei der Bearbeitung von Workflow-Aktivitäten. Die Informationsbedürfnisse der Bearbeiter für jede Aktivität werden dabei grundsätzlich im Prozessmodell durch die Spezifikation von sogenannten wissensintensiven Aktivitäten festgelegt. **FRODO** (Framework for Distributed Organizational Memories, [BDE⁺04]) ist die Fortführung des KnowMore-Projektes und ergänzt den Ansatz um zusätzliche Werkzeuge. Das **DECOR**-Projekt (Delivery of Context-Sensitive Organizational Knowledge) überträgt den Ansatz auf schwachstrukturierte Workflows ([ABD⁺01]).

- Die Grundidee des **AD-HOC**-Systems ([Far03], [FLDL04], [LF04]) des Grazer Know-Center ist die Nutzung von Wissensmanagementsystemen für das Lernen, ohne dass die Inhalte speziell didaktisch aufbereitet werden müssen. Es bietet auf der Basis des Hyperwave-Content-Management-Systems einen prozessorientierten, reaktiven Zugang.
- Der **KontextNavigator** ([Goe01b], [Goe01a], [DGHH02]) vom Fraunhofer Institut für Software- und Systemtechnik ist ein prozessorientiertes System zur Verwaltung des organisationalen Gedächtnisses, das als Erweiterung eines Workflow-Management-Systems konzipiert und realisiert wurde. Das System versteht sich als Unterstützungsfunktion bei der Prozessbearbeitung. Der Mitarbeiter kann das System direkt aus der workflowgetriebenen Anwendung heraus aufrufen. Ausgehend von seinem aktuellen Prozesskontext kann er dann z.B. entlang von Prozessen im organisationalen Gedächtnis navigieren. Dem Mitarbeiter ist es dabei nicht nur möglich, prozessorientiert Dokumente abzurufen, sondern auch Dokumente ins System einzustellen, wobei auch komplexere Sichtbarkeitsbedingungen auf der Basis von Workflow-Variablen möglich sind.

Die hier vorgestellten Ansätze der *geschäftsprozessorientierten Informationsversorgung* versuchen alle, Lernen (meist als »Wissenserwerb« bezeichnet) in die Arbeitsprozesse zu integrieren. Das Hauptinstrument ist hierbei die Beschreibung der Arbeitssituation durch die jeweiligen Geschäftsprozesse und die entsprechende Zuordnung der Artefakte zu diesen Prozessen. Damit wird die Arbeitssituation im wesentlichen reduziert auf den jeweiligen Prozessschritt. Eine darüber hinausgehende Berücksichtigung der Situation findet kaum oder allenfalls beschränkt auf die konzeptionelle Ebene statt. Proaktive Unterstützung ist schon in den ersten Projekten am DFKI ein wesentliches Element, die den Übergang zwischen Arbeiten und Lernen auch operationalisiert, obwohl man proaktive Funktionalität nicht in allen hier vorgestellten Projekten findet. Pädagogische Führung – im weiteren Sinne – ist durchaus in den Ansätzen enthalten. Insbesondere der Ad-Hoc-Ansatz mit der Bereitstellung unterschiedlich aufbereiteter Inhalte und der Wissenslage-Ansatz mit dem Potential, unterschiedliche Arten von Aktivitäten bereitzustellen, gehen in diese Richtung. Allerdings ist diese Führung nicht pädagogisch motiviert und auch sehr eingeschränkt. Kennzeichnend gegenüber den E-Learning-Ansätzen ist die pragmatische Herangehensweise, bei der (wenn überhaupt) nicht ausschließlich auf didaktisch aufbereitetes Material zurückgegriffen wird, sondern stärker auf Gebrauchsdokumente, was gerade in Unternehmenskontexten wichtig ist.

4.4. Fazit

In diesem Kapitel wurde der Fokus des weiteren Verlaufs der Arbeit umrissen: es geht um die Erarbeitung einer Methodik für eine neue Steuerungsform von Lernprozessen, die speziell auf das Paradigma eines Lernens bei Bedarf ausgelegt ist, und um eine technische Infrastruktur, die diese Methodik umzusetzen vermag. Nachdem wir bereits in diesem Kapitel auf methodische Ansätze aus dem Bereich der Pädagogik kurz eingegangen sind soll im folgenden Kapitel noch die Informationsrecherche analysiert werden, bevor wir die Methodik als Synthese erarbeiten.

5.

Informationsrecherche als Lernprozess

Gerade für das Lernen bei Bedarf ist die Informationssuche besonders dominierend; sie kann – wie in diesem Kapitel gezeigt wird – als ein selbstgesteuerter Lernprozess bzw. als ein Element von selbstgesteuerten Lernprozessen aufgefasst werden. Will man diese Lernprozesse verbessern, so muss man die Informationssuche an sich bzw. ihre Einbettung in den gesamten Lernprozess verbessern. Deshalb soll in diesem Kapitel Modelle für den Suchprozess gezielt dahingehend analysiert werden, welche Bezüge sie zu Lernprozessen aufweisen. Gleichzeitig dient das Kapitel als Grundlage für die Erarbeitung der Methode des »situationsgesteuerten Lernens«, indem Interventionsmöglichkeiten aufgezeigt werden.

5.1. Grundbegriffe

5.1.1. Suche und Recherche

In vielen Fällen werden die Begriffe »Informationssuche« und »Informationsrecherche« als austauschbar betrachtet oder in unterschiedlichen Schattierungen verwendet. Im folgenden soll daher kurz skizziert werden, wie sich die Begriffe gegeneinander abgrenzen lassen, indem man sich menschliches Verhalten vornimmt ([Wil99], vgl. auch [Lit04]):

- **Informationsverhalten** (*information behavior*). Dies bezeichnet alle Aktivitäten zur Identifikation des Informationsbedürfnisses, zur Suche nach passenden Informationen und zur Nutzung und Übertragung der gefundenen Information auf das eigene Problem.
- **Informationsrechercheverhalten** (*information seeking behavior*). Darunter fallen alle Aktivitäten, die eingesetzt werden, um Informationsquellen zu entdecken und zu ihnen Zugriff zu erhalten.

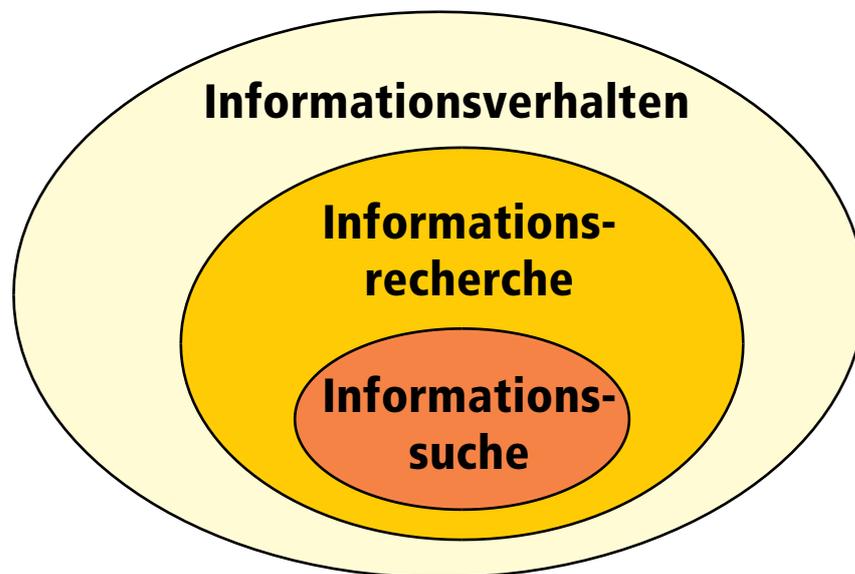


Abbildung 5.1.: Informationsverhalten, Informationsrechercheverhalten und Informationssuchverhalten nach [Wil99]

- **Informationssuchverhalten** (*information search behavior*). Dieses bezieht sich auf die Interaktionen zwischen dem Benutzer und rechnergestützten Informationssystemen.

5.1.2. Informationsbedürfnis

Der Startpunkt für die Informationssuche ist das sog. »Informationsbedürfnis«, das als Begriff durch [Tay68] als ein persönlicher psychischer Zustand eingeführt wurde, der oft vage und unbewusst ist. Er differenziert nach vier verschiedenen Ebenen des Bedürfnisses:

- **Latentes Bedürfnis** (*visceral need*). Ein latentes Bedürfnis wird dem Betroffenen noch nicht bewusst, sondern drückt sich allenfalls in Emotionen aus, die noch nicht gedeutet werden.
- **Bewusstes Bedürfnis** (*conscious need*). Der Betroffene wird sich bewusst, dass sein derzeitiger Wissens- bzw. Informationsstand nicht ausreicht.
- **Formalisiertes Bedürfnis** (*formalized need*). Das Bedürfnis wird sprachlich ausdrückbar in Form von Suchphrasen.
- **Angepasstes Bedürfnis** (*compromised need*). Die Formalisierung des Bedürfnisses passt sich an die Möglichkeiten eines konkreten Informationssystems an, das für

seine Befriedigung benutzt wird. Dies umfasst die Terminologie und Sprache, die Möglichkeiten der Anfragesprache bzw. Navigationsmöglichkeiten etc.

Erst die letzte Ebene, bei der das Bedürfnis an die Möglichkeiten des Informationssystems angepasst wurde, erlaubt eine Interaktion (sprich: eine Anfrage) mit dem System.

Aus dieser Einteilung des Bedürfnisses in unterschiedliche Ebenen lassen sich mehrere Schlussfolgerungen ziehen, die für die vorliegende Arbeit von Bedeutung sind:

- Bevor ein Nutzer aktiv ein Informationssystem zur Befriedigung seines Informationsbedürfnisses konsultiert, muss er eine recht große kognitive Leistung aufwenden, um das Bedürfnis in eine Anfrage zu übersetzen. Deshalb wird in alltäglichen Nutzungssituationen wohl nur ein Bruchteil der latenten oder bewussten Informationsbedürfnisse auch tatsächlich in Interaktionen mit Informationssystemen umgesetzt.
- Um latente oder auch nur bewusste Informationsbedürfnisse zu befriedigen, muss das System versuchen, das Formalisieren und Anpassen vorwegzunehmen, was sicherlich nur auf unvollkommene Art und Weise geschehen kann.

Das Informationsbedürfnis entsteht typischerweise in einer Situation, in der der Betroffene subjektiv wahrnimmt, dass sein aktuelles Wissen oder Problemlösevermögen für eine erwartete Problemlösung nicht ausreicht. Für diese Situation existieren unterschiedliche Bezeichnungen wie »anomaler Wissenszustand« (anomalous state of knowledge, ASK) [Bel80], »problematische Situation« ([Lin01], [KB02], [LB05]) oder »Kontext der Informationsrecherche« [Mar95]. Die Charakterisierung dieser Situation ist noch immer Gegenstand von Forschungsaktivitäten, wie z.B. in [KB02], wo empirisch hierfür als Dimensionen thematische Vertrautheit, Fortdauer des Themas, Dauer der Aufgabe und Stadium des Problemlöseprozesses ermittelt wurden.

5.1.3. Akteure und ihre Aktionen

Um die Vielgestaltigkeit der Informationsrecherche besser zu strukturieren, hat WILSON unterschiedliche Suchpfade identifiziert (vgl. Abb. 5.2), die im folgenden bezeichnet und in Zusammenhang mit dem obigen Wissensreifungsprozess und der darin strukturierten Lernformen gebracht werden sollen. Der grundsätzliche Aufbau ist dabei dreigeteilt: links der Benutzer und seine soziale Umwelt, in der Mitte das Informationssystem, rechts die Informationsressourcen. Bei den Kreisen wird ein bestimmtes Umfeld unterstellt, d.h. ein Benutzer sucht in seiner beruflichen Rolle. Dementsprechend ist seine Referenzgruppe dann seine Abteilung oder die Kollegen im Projekt. Ähnlich wird davon ausgegangen, dass ein bestimmtes Informationssystem bereitgestellt wird, das einen Informationsbestand erschließt.

- a) **Konsultation der Referenzgruppe.** Hierbei handelt es sich um informelle Lernprozesse in der Referenzgruppe, die üblicherweise die unmittelbaren Kollegen umfasst. Die Barrieren sind hier relativ niedrig, weshalb auch relativ »unreife« Informationen ausgetauscht werden können.
- b) **Konsultation der Lebenswelt.** Diese Suchpfade erweitern den Kreis der sozialen Interaktionen auf die weitere Lebenswelt. Wegen des oft fehlenden gemeinsamen Kontextes können hierüber nur reifere Informationen ausgetauscht werden.
- c) **Konsultation der restlichen Welt.** Hier ist durch die fehlende (bestehende) soziale Beziehung der Informationsaustausch noch weiter erschwert.
- d) **Anfrage an menschlichen Mediator.** Der Unterschied zur sozialen Interaktion ist die Professionalisierung der Beziehung. Hierbei wird eine Rollenverteilung unterstellt, in der der Mediator die Aufgabe hat, bei der Informationsrecherche zu helfen. Dieser Mediator kann im Sinne des Lernens auch ein Tutor sein, der das Lernen unterstützt und an geeigneten Stellen Hilfestellungen anbietet. Je nach Nähe des Mediators ist ein gewisser Reifegrad unabdingbar.
- e) **Anfrage an technisches Informationssystem.** Die Mediationsrolle kann auch von einem technischen System übernommen werden. Hier kommen dann allerdings nur Lernformen ins Spiel, die auf Artefakten basieren.
- f) **Direkter Zugriff auf Informationsressourcen.** Dies ist beispielsweise bei Portfolios möglich, in denen der Lernende bereits die Dinge so vorstrukturiert hat, dass er direkt das Gesuchte findet. Auch sog. »ready-reference«-Situationen werden hiervon abgedeckt.
- g, h, i) **Nutzung von Informationsquellen durch den Mediator.** Zur Dienstleistung kann der Mediator auf das technische System, externe Quellen oder direkt auf die Ressourcen zugreifen.
- j,k) **Zugriff des technischen Systems auf Informationsquellen.** Das technische System kann sowohl externe Quellen als auch die betrachtete Menge an Informationsressourcen zugänglich machen.

Was interessanterweise in dieser Systematik fehlt, ist jegliche Initiierung von außen, sei es durch das soziale Umfeld, den menschlichen Mediator oder das technische System. Hier zeigt sich eine der grundsätzlichen Einschränkungen der gesamten Informationsverhaltensforschung.

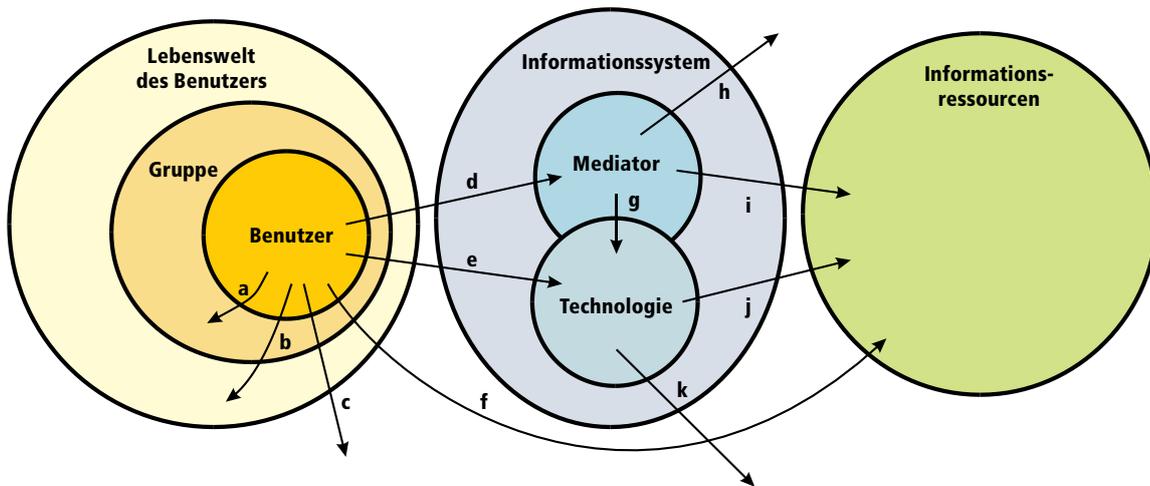


Abbildung 5.2.: Wilsons Kontext der Informationssuche [Wil81]

5.1.4. Relevanz

Der Begriff »Relevanz« hat sich als einer der zentralen Begriffe des Information Retrieval etabliert [Sar96]. Das Ziel von IR-Systemen ist das Liefern von »relevanten« Ergebnissen. Relevanz wird hierbei allgemein verstanden als eine Beziehung, die ausdrückt, dass die eine Seite einen direkten Einfluss auf die andere Seite hat [TOKF05]. Die Relevanz eines Dokumentes als Gütemaß ist also stets bezogen auf eine Anfrage oder auf die Situation einer Person. Wichtig ist hierbei zwischen einem objektiven und einem subjektiven Relevanzbegriff zu unterscheiden: Während objektive Relevanz nur die explizite Anfrage als Grundlage nimmt und davon abstrahiert, was der Benutzer tatsächlich meinte [Swa86], betrachtet der subjektive Relevanzbegriff Relevanz im Bezug auf den Wissensstand und das Informationsbedürfnis des Benutzers in der jeweiligen Situation ([IB96], [Sar96], [MS02a], [LTGF04], [TOKF05]).

Dies stützt sich auf die Sichtweise des Symbolischen Interaktionismus, wie sie maßgeblich von BLUMER verkörpert wird. Die wesentlichen Thesen sind hierbei [Blu69]: (1) Der Mensch handelt gegenüber den Dingen aufgrund der Bedeutung der Dinge. (2) Die Bedeutung der Dinge entsteht aus der sozialen Interaktion. (3) Die Bedeutungen werden durch einen interpretativen Prozess, den die Person in ihrer Auseinandersetzung mit den ihr begegnenden Dingen benutzt, gehandhabt oder abgeändert. Daraus ergibt sich, dass Informationsartefakte keine Bedeutung (und damit auch keine Relevanz an sich haben), sondern diese erst in einem interpretativen Prozess erarbeitet wird. Dieser Interpretationsprozess ist dabei Teil eines größeren sozialen Aushandlungsprozesses, so dass ein und dasselbe Artefakt in unterschiedlichen sozialen Kontexten unterschiedliche Bedeutungen

haben kann.

Daraus dürfte klar sein, dass für den praktischen Nutzen eines IR-Systems einzig und allein die subjektive Relevanz ausschlaggebend ist, doch hat objektive Relevanz den Vorteil, dass sie sich ohne Benutzerstudien ermitteln lässt und sich so ein System leichter objektiv evaluieren lässt, so dass sie in der IR-Forschung vorherrscht.

Was subjektive Relevanz zudem schwierig zu operationalisieren macht, ist ihre Vielschichtigkeit. Deshalb gab es zahlreiche Versuche, sie weiter in unterschiedliche Bestandteile zu zerlegen. Die Klassifikation in »Manifestationen von Relevanz« von SARACEVIC ist hierzu repräsentativ ([Sar96], vgl. auch [TOKF05]):

- **Kognitive Relevanz.** Dies bezeichnet die Beziehung zwischen dem Wissensstand bzw. dem Informationsbedürfnis des Benutzers und Informationsartefakten. Hierzu gehört z.B. die Möglichkeit, die Dokumente zu verstehen und mit ihrer Hilfe neues Wissen zu konstruieren.
- **Situative Relevanz.** Auch als »Nützlichkeit« bezeichnet, betrachtet dieser Relevanzbegriff die Beziehung zwischen Informationsartefakten und der Situation (was z.B. auch die Aufgabe oder das aktuelle Probleme beinhaltet). Wichtig hierbei ist, dass dies aus der subjektiven Perspektive des Benutzers zu beurteilen ist [Wil73], also z.B. der tatsächlich wahrgenommene Nutzen für ein Entscheidungsproblem.
- **Motivationale/affektive Relevanz.** Dies beschreibt die Relevanz zwischen Informationsartefakten und den Absichten, Zielen, Motivationsfaktoren eines Benutzers, was sich in Zufriedenheit oder Erfolg niederschlagen kann. Hierzu dürfte auch WILSONS »psychologische Relevanz« zu rechnen sein, die sich mit den Auswirkungen der Nutzung der Information auf das Verhalten der Person beschäftigt [Wil73].

Die unterschiedlichen »Manifestationen« sind dabei nicht trennscharf (und auch nicht so gedacht [Sar96]), sondern sollen stattdessen den Raum der subjektiven Relevanz konkretisieren. Diese Ergebnisse liefern eine wichtige Grundlage für die Konkretisierung und Operationalisierung unseres Situationsbegriffes (wofür im Gegensatz zur engen situativen Relevanz alle drei obigen Relevanzarten zu betrachten sind).

5.2. Modelle für das Informationsverhalten

5.2.1. Wilsons Modell des Informationsverhaltens

Das in dieser Arbeit betrachtete Modell für das Informationsverhalten von WILSON (vgl. [Wil81]) konzentriert sich auf den Übergang von einem in der Situation verhafteten Informationsbedürfnis hin zu einem Informationsrechercheverhalten. Insbesondere interessieren WILSON die Barrieren, die einen Eintritt in eine Informationsrecherche verhindern.

Als Einflussvariablen, die sowohl die oben erwähnten Barrieren konstituieren, als auch den weiteren Verlauf beeinflussen, identifiziert WILSON in [WW96] auf der Basis einer breiten Analyse der empirischen Erkenntnisse vor allem aus der Psychologie die folgenden:

- Persönliche Eigenschaften
 - Kognitive Dissonanz als Motivation, einen Zustand widersprüchlicher Wahrnehmungen aufzulösen
 - Selektive Zuwendung (*selective exposure*) als Befangenheit zugunsten von erwartungskonformen Informationen
 - Ausbildungs- und Wissensstand
- Emotionale Einflußgrößen
- Demographische Einflußgrößen (Alter, Geschlecht)
- Soziale/interpersonale Einflußgrößen
- Situationelle und Umwelteinflußgrößen (z.B. Zeitdruck, Ort, aber auch Kultur)
- Ökonomische Einflußgrößen (z.B. Preise für Recherche)
- Quelleneigenschaften (Zugangsmöglichkeiten, Glaubwürdigkeit, Kommunikationskanal)

WILSON [WW96] identifiziert bei einer genaueren Analyse der Aktivierungsmechanismen zwei Ebenen, die entscheiden, ob überhaupt die Suche gestartet wird. Auf der ersten Ebene spielt das Ausmaß von Stress und die Fähigkeit des Individuums, neues Wissen einzubauen, eine große Rolle. Dies wird von den psychologischen »Stress-Coping-Theorien« abgedeckt. Dabei bezeichnet »coping« einen Zustand, in denen das Individuum in der Lage ist, mit Hilfe der eigenen Wissenstrukturen neues Wissen aufzubauen und dementsprechend auch dazu motiviert ist [FL85], während Stress ein Zustand ist, in dem neues Wissen nicht willkommen ist, da es die Unsicherheit bzgl. des eigenen Wissens noch vergrößern würde. Ein gestresstes Individuum wird daher versuchen, alles was seine Fähigkeiten zum Aufbau weiterer Wissenstrukturen zu überfordern droht, zu ignorieren. Auf der zweiten Ebene schätzt das Individuum ab, ob die Informationsrecherche wohl zum Erfolg führen wird, wozu sog. »Risk-Reward-Theorien« herangezogen werden. Erfolg ist dabei subjektiv zu sehen: eine Recherche ist dann erfolgreich, wenn sich das Individuum durch den Verlauf und/oder Ausgang in seiner Kompetenz bestätigt sieht.

Auf der ersten Ebene kann die Proaktivität eines Systems helfen, den Stress zu reduzieren, indem keine Formalisierung des Bedürfnisses abverlangt wird. Auf der zweiten Ebene kommt es darauf an, Frustrationen zu vermeiden.

5.2.2. Modell des Informationsverhaltens nach Niedzwiedzka

NIEDZWIEDZKA präsentiert ein Modell (in Modifikation eines Modells von WILSON), das sowohl direkte, auf sich selbst gestellte Informationsrecherche, als auch die Inanspruchnahme eines Mediators (z.B. Bibliothekspersonal) umfasst. Es beginnt mit der Identifikation des Informationsbedürfnisses durch den Benutzer und einer Entscheidung, eine Informationsrecherche durchzuführen. Die Recherche kann dann mit Hilfe eines Mediators oder eigenständig durchgeführt werden, wobei unterschiedliche Hilfsmittel wie eigene Informations- und Literatursammlungen, Informationssysteme oder Bibliotheken genutzt werden (sowohl durch Mediator als auch durch den eigenständigen Nutzer). Als nächster Schritt wird die gefundene Information verarbeitet und anschließend angewendet, wobei im Rahmen der Anwendung wiederum festgestellt werden kann, dass ein noch unbefriedigtes Informationsbedürfnis vorliegt.

An dem Modell lässt sich sehr gut ablesen, wo kritische Punkte innerhalb des Prozesses sind, an denen sich auch intervenieren lässt (vgl. Abb. 5.3) [Nie03], weil dort »Aktivierungsmechanismen« eine Rolle spielen:

- **Identifikation des Informationsbedürfnisses.** Wie oben ausgeführt ist das Bedürfnis zunächst einmal latent. Ob es bewusst wird, hängt auch von kontextuellen Faktoren ab.
- **Entscheidung zur Informationsrecherche.** Ob eine Informationsrecherche durchgeführt wird, hängt davon ab, inwieweit der Benutzer es sich zutraut, genügend zu wissen und wie groß die Belastung ist (da Informationsrecherche zu einer Erhöhung der Belastung führt). An dieser Stelle kommen die oben erwähnten Theriefamilien »Stress/Coping« und »Risk/Reward« zum Tragen. Hier können frühere Frustrationen oder Erfolgserlebnisse eine große Rolle spielen.
- **Durchführung der Informationsrecherche.** Der tatsächliche Verlauf der Informationsrecherche hängt offensichtlich sehr stark von persönlichen und anderen kontextuellen Faktoren ab.
- **Auswahl und Verarbeitung der Information durch den Benutzer.** Auch bei der Auswahl und Verarbeitung, was letztendlich nichts anderes als ein Lernprozess ist, spielen zahlreiche kontextuelle Faktoren eine Rolle.

Dieses Modell identifiziert klar die Interventionspunkte, an denen ein System unterstützend wirken kann, wenn auch die Wirkungszusammenhänge (was hilft, Barrieren zu überwinden; was wirkt unterstützend?) von diesem Modell nicht geliefert werden. Hier sind insbesondere die ersten beiden Interventionspunkte vielversprechend. Bei der Identifikation des Bedürfnisses kann das System zur Bewusstwerdung beitragen (z.B. durch

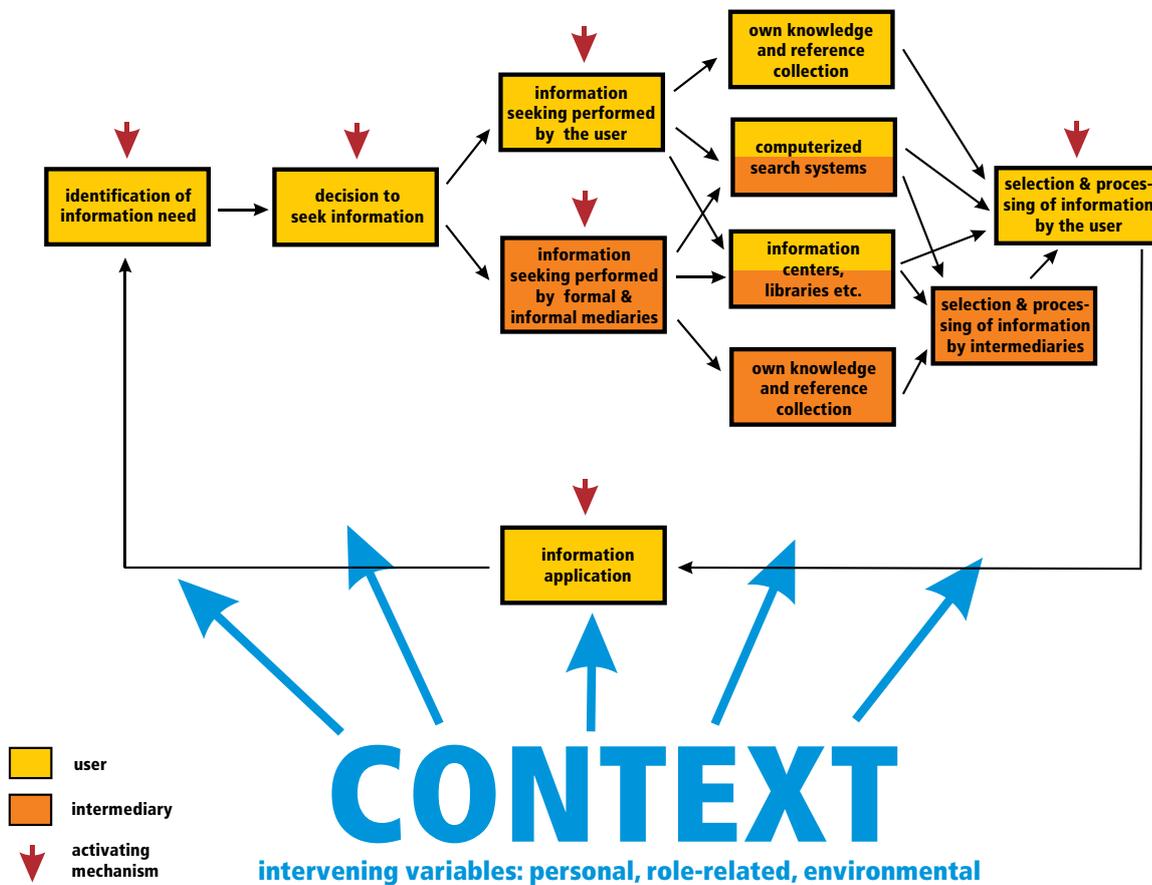


Abbildung 5.3.: Niedzwiedzka's Modell des Informationsverhaltens[Nie03], aufbauend auf [WW96]

Signalisierung). Bei der Entscheidung zur Informationsrecherche ist das Wirkungsfeld einer Lernunterstützung durch proaktive Empfehlungen noch größer. Durch Vorwegnahme einer ersten Informationsrecherche (das Ergebnis sind hier die Empfehlungen) wird die Anfangsbelastung reduziert; zudem werden Frustrationen durch leere Ergebnismengen vermieden. Achtet das System in der Durchführungsphase zudem noch darauf, dass dem Benutzer die Dinge so präsentiert werden, wie er sie auch verstehen kann, wird vermieden, dass er die Durchführung abbricht und Überforderung mit Rechercheaktivitäten verbindet.

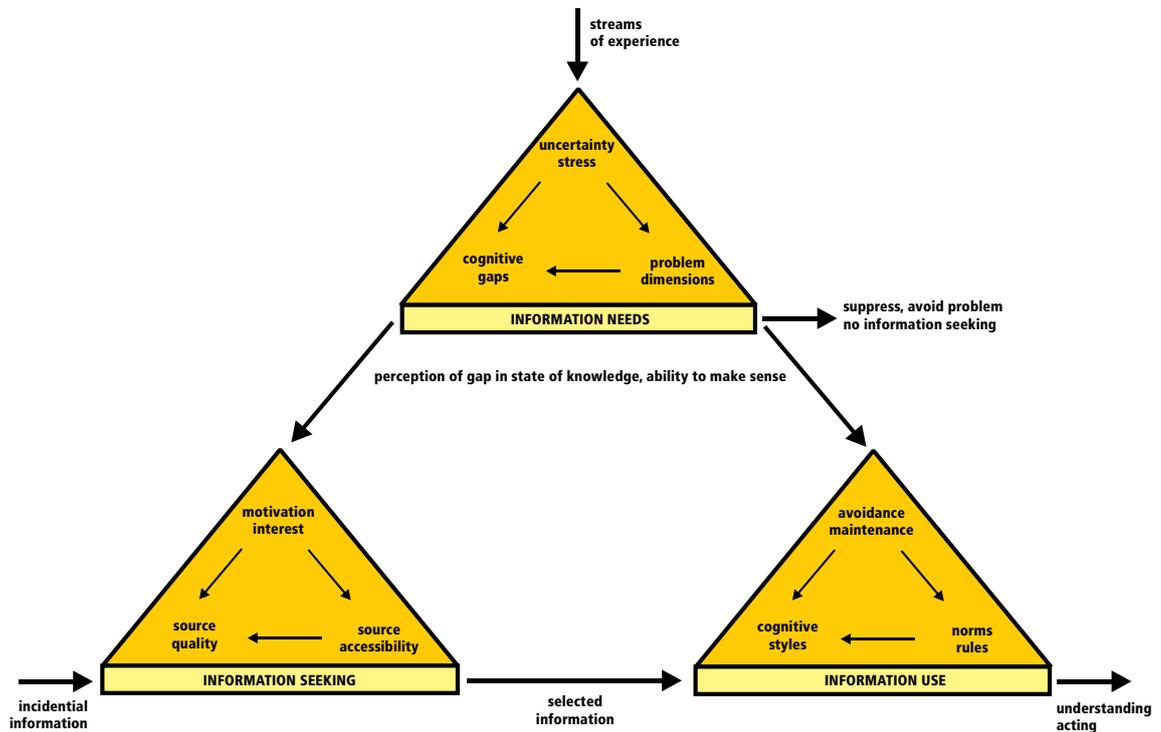


Abbildung 5.4.: Informationsrecherche nach Choo [Cho02]

5.2.3. Informationsverhaltensmodell nach Choo

CHOO [Cho02] ordnet das Informationsverhalten in ein Dreieck aus Informationsbedürfnissen, Informationsrecherche und Informationsnutzung (vgl. Abb. 5.4). Das Individuum verspürt ein Bedürfnis, wenn eine Lücke im Wissensstand oder der Fähigkeit, Verständnis aufzubauen, wahrgenommen wird. Die Wahrnehmung wird durch kognitive, affektive und situationelle Faktoren beeinflusst. Das Bedürfnis kann unterdrückt werden, oder es wird versucht, die Lücke durch eine zweckorientierte Informationsrecherche zu schließen. Die Auswahl und Nutzung von Quellen während der Recherche wird beeinflusst von subjektiver Zugänglichkeit und Qualität, von der Aufgabenkomplexität und den persönlichen Interessen. Dabei kann auf Informationen auch zufällig gestoßen werden, z.B. durch gewohnheitsmäßiges Verfolgen der Fachpresse oder Unterhaltungen mit anderen.

Das Ergebnis der Informationsrecherche ist eine Menge von ausgewählten Informationen, die dann verarbeitet und angewandt werden, was wiederum beeinflusst wird durch den Kognitionsstil, die Emotionen während der Informationsverarbeitung und die sozio-

kulturelle Umgebung. Das Resultat ist eine Veränderung im individuellen Wissensstand, der das Individuum in die Lage versetzt, zu verstehen oder zu handeln. Diese »Veränderung im Wissensstand« bzw. der Erwerb von Handlungskompetenz ist nichts anderes als das Ergebnis eines Lernprozesses, so dass in diesem Modell sehr klar die Bezüge herausgestellt werden.

Wichtig an diesem Modell ist die deutliche Abgrenzung zwischen Informationsbedürfnis, Informationsrecherche und Informationsnutzung, wobei kein linearer Prozess angenommen wird, so dass es unterschiedliche Pfade geben kann. Für die vorliegende Arbeit ist dabei besonders wichtig, dass es zwischen Bedürfnis, Recherche und Nutzung Prozessbarrieren gibt. Hierbei ist besonders die Unterdrückung des Informationsbedürfnisses von Interesse.

Interessant an dem Modell ist außerdem, dass es auch explizit auf Arbeitssituationen (auf das Informationsverhalten von Führungskräften) angewandt wurde [Cho01] und sich nicht auf das klassische Bibliotheksszenario beschränkt. Dabei identifiziert CHOO besonders auch affektive (Kontext-)Faktoren wie Unsicherheit (vgl. hierzu auch die im folgenden Abschnitt vorgestellte Theorie von KUHLTHAU), (intrinsische) Motivation, aber auch Umgebungsfaktoren wie Normen und Regeln.

5.3. Modelle für das Informationsrechercheverhalten

5.3.1. Kuhlthaus Informationsrechercheprozessmodell

Das Informationsrechercheprozessmodell¹ von Kuhlthau [Kuh04] ist kein reines Verhaltensmodell, sondern beschreibt die Informationsrecherche auf ganzheitliche Art und Weise, was insbesondere auch die emotionale Seite mit einschließt. Ausgehend von einer konstruktivistischen Lerntheorie entwickelt KUHLTHAU eine Theorie über die Anatomie des Informationsrechercheprozesses, deren Leitidee die Ungewißheit (*uncertainty*) ist. Ungewißheit wird dabei als kognitiver Zustand verstanden, der durch vage Gedanken und verunsichernde Gefühle (*anxiety*) gekennzeichnet ist und sich bei wahrgenommenen Verständnislücken (*gaps of knowledge*) offenbart. Dabei entsprechen der Ungewißheit auf der Ebene des Denkens die gedankliche Unklarheit, auf der Ebene der Gefühle die Unsicherheit und Angst und auf der Ebene des Handelns das Entdecken (im Gegensatz zum Sammeln/Dokumentieren).

Das wichtigste Element dieser Theorie ist die Feststellung, daß die Informationsrecherche ein Prozess ist, der anhand charakteristischer Merkmale aus den drei Lernbereichen (kognitiv, affektiv, psychomotorisch) in sechs Phasen aufgeteilt werden kann, wobei die

¹KUHLTHAU spricht zwar selbst von »Informationssuchprozess«, in der Begriffsbildung wie oben vorgestellt ist »Informationsrechercheprozess« allerdings treffender. Dies wird auch dadurch gestützt, dass KUHLTHAU selbst des öfteren von »Informationsrecherche« spricht.

lineare Abfolge nicht impliziert ist, sondern ausdrücklich von zyklischen Strukturen ausgegangen wird. Innerhalb des Prozesses ist generell von einer Stufe zur anderen von einer Abnahme der Unsicherheit auszugehen.

- **Phase 1: Vorbereitung (task initiation).** Man erkennt, daß um eine gegebene Aufgabe oder ein Problem zu lösen, Information benötigt wird, über die man nicht verfügt. In der Vorbereitungsphase geht es im wesentlichen um das Verstehen der Aufgabe und das Vergleichen mit bereits gelösten Problemen, aber auch um das Nachdenken und Diskutieren über mögliche Themen der Informationsrecherche.
- **Phase 2: Themenauswahl (topic selection)** In dieser Phase fällt die Entscheidung, welches Thema man durch seine Informationssuche bearbeiten will. Hierzu wird eine vorläufige und überblicksartige Suche durchgeführt, die mögliche Alternativen zu bestimmen und gegeneinander abzuwägen versucht.
- **Phase 3: Orientierung (pre-focus exploration).** Nachdem man sich auf ein Thema festgelegt hat, geht es darum, sich in dem gewählten Thema zu orientieren und sich die dazu notwendigen Kenntnisse anzueignen. Vom kognitiven Standpunkt aus ist dies die schwierigste Phase, da hier versucht werden muss, in der Regel inkompatible Informationen in das vorhandene Wissen zu integrieren. Man ist offen für viele Aspekte und neue Denkansätze.
- **Phase 4: Schwerpunktformulierung (focus formulation).** Kritisch für den weiteren Verlauf ist das Festlegen eines Schwerpunktes. Dies geschieht durch Auswahl von Ideen, Ansätzen und Aspekten, auf die man sich konzentrieren möchte. Hier werden oftmals Thesen formuliert, die im weiteren als Leitsätze fungieren.
- **Phase 5: Sammeln von Information (information collection).** Nachdem die Klarheit über den Schwerpunkt der eigenen Arbeit gewonnen ist, werden nun gezielt Informationen gesammelt. Dabei werden Ideen, Thesen oder Ansätze der vorangegangenen Phase gezielt präzisiert und überprüft.
- **Phase 6: Abschluß (search closure).** In dieser Phase wird die Suche abgeschlossen, indem die gefundenen Informationen meist in einer eigenen Arbeit zusammengefasst werden.

KUHLTHAU basiert ihr Informationsrechercheprozessmodell auf der konstruktivistischen Lerntheorie und legt damit die Basis, Informationssuche als eine spezielle Form von Lernprozess zu verstehen. Es liegt hierbei implizit das Szenario einer Seminar- oder Diplomarbeit zugrunde, das als Ziel die Dokumente zu einem bestimmten Thema hat. Dieses Modell findet sich freilich nicht nur in formellen Lernsituationen wieder, sondern

ist oft auch charakteristisch für sog. »Wissensarbeit«, bei der bestimmte Inhalte für Berichte, Artikel, Präsentationen o.ä. aufbereitet werden müssen. Hier kann man so weit gehen, dass der Arbeitsprozess untrennbar mit dem Lernprozess verbunden ist: die Arbeitssituation ist eine Lernsituation. Damit verwischen hier auch die Grenzen zwischen Arbeitswerkzeugen und Lernunterstützung.

Für Information-Retrieval-Anwendungen ist bedeutsam, dass die Ausrichtung am Anfrage-Ergebnis-Paradigma und die Grundannahme, dass in der Anfrage das Informationsbedürfnis spezifiziert ist, durch den klaren Prozesscharakter des Suchens der realen Welt nicht gerecht werden. Dies wird auch durch die beiden Suchprozeßmodelle *orientierung* [OJ93] und *berry-picking* ([Bat89], [Bat02]) gestützt, die nahelegen, dass ein einziger Suchprozess sich normalerweise über unterschiedliche Interaktionen und evtl. auch Dienste hinweg erstreckt, weil das »normale« Ergebnis nicht eine einzige Ergebnismenge, sondern eine Reihe von Informationshäppchen darstellt, die es zusammenzufügen gilt. Das Verbindende zwischen diesen einzelnen Interaktionen ist ein individueller Lernprozess, für den dementsprechend auch die Erkenntnisse über Lernprozesse zu gelten haben. Will man die Informationssuche verbessern, so kann man das nur, indem man den Lernprozess besser unterstützt – und nicht, indem man versucht, die ideale Ergebnismenge zu präsentieren.

Weiterhin ergibt sich schon allein aus der Tatsache, dass KUHLETHAU ihr Modell entwickelt hat, um die Interventionsmöglichkeiten im Suchprozess durch Bibliothekare zu systematisieren, dass die komplette Selbststeuerung von Suchprozessen nur die zweitbeste Lösung ist. Stattdessen sprechen insbesondere die affektiven Beobachtungen dafür, dass Suchprozesse geführt werden sollten. Dies stellt nichts anderes dar als die pädagogische Grundannahme, dass sich Lernen durch (pädagogisches) Führen effizienter gestalten lässt. Dieses Führen kann sowohl im zwischenmenschlichen Bereich, als auch durch das System erfolgen.

Aus der Perspektive des Lernens ist ein besonders zu berücksichtigender Punkt, dass der Such-/Lernprozess ganzheitlich zu betrachten ist und man insbesondere die affektive Seite nicht ignorieren darf. Hier liegen wichtige Einsichten für Lernumgebungen für das selbstgesteuerte Lernen vor: Der Benutzer ist nicht immer Herr seiner Lage, der selbstsicher das Recherche-/Lernsystem als Werkzeug einzusetzen vermag. Vielmehr ist vor allem in den Anfangsphasen des Suchprozesses mit Unsicherheit zu rechnen, die ein System nicht noch durch Überforderung des Benutzers durch zu rigide Anforderungen an die Anfrageformulierung verstärken darf. Insbesondere ist auch davon auszugehen, daß der Benutzer (vor allem in Phase 3) überhaupt nicht weiß, was er sucht/lernen möchte (oder es zumindest nicht ausdrücken kann). Hier kann ein System lernförderlich wirken, indem es die Formalisierung des Informations- oder Wissensbedürfnissen durch Situationsbewusstsein antizipiert und entsprechende Ressourcen proaktiv vorschlägt.

Kuhlthau weist in ihrer Theorie auch darauf hin, dass im Rahmen der Suche ein Gleichgewicht zwischen Redundanz und Einzigartigkeit gewährleistet sein muss. Erhält ein Be-

nutzer als Ressourcen nur »Neues« oder »Einzigartiges« (ist also jegliche Redundanz eliminiert), so fühlt er sich verloren, und es erhöht seine Unsicherheit. Ein gewisses Maß an Redundanz schafft Vertrautheit, die sich förderlich auf den Lernprozess auswirkt, da Anknüpfungspunkte existieren. Allerdings führt zuviel Redundanz zu Langeweile; das Neue muss sich auch klar vom Bekannten abheben und so wahrgenommen werden können. Hieraus ergibt sich, dass es von essentieller Bedeutung ist, dass bei der Bereitstellung von Ressourcen (sei es reaktiv oder proaktiv) das Vorwissen und die Historie des Benutzers miteinbezogen wird.

5.3.2. Wang Baldonados Anwendung der Sensemaking-Theorie auf die Informationsuche

Um zu einem konzeptuellen Modell für die Bereitstellung von suchunterstützenden Funktionen (wie z.B. Strukturierungsoperationen für die Ergebnismengen oder kontextsensitive Folgeanfragen) zu gelangen, stützt sich WANG BALDONADO auf die »Sensemaking«-Theorie (vgl. z.B. [Der99]). Die Idee des »Sensemaking« ist hierbei, dass Individuen Informationen suchen, um ein Verständnis für die Situationen in ihrem Leben aufzubauen, indem Erfahrungen und Ideen so strukturiert werden, dass es dem Streben nach Ganzheit des Individuums entspricht. Für die Theorie sind konzeptionell vier wesentliche Elemente auszumachen [Wil99]: die *Situation*, in der das Informationsproblem entsteht, die *Lücke* als Unterschied zwischen tatsächlicher und gewünschter Situation, die *Brücke* als Mittel zum Schließen der Lücke und das *Ergebnis* als Folge des Sensemaking-Prozesses.

WANG BALDONADO entwickelt ein Modell für die Mikroschritte eines Suchprozesses [WB97]. Sie geht von einem Informationsbegriff aus, der besagt, daß Information nicht einfach durch Kodierung/Dekodierung übertragen werden kann, sondern nur durch dialogische Prozesse, die auf beiden Seiten auf Kontext und Vorverständnis aufsetzen. Dies entspricht im wesentlichen der bereits im Rahmen der Definition des Wissensbegriffes kurz aufgegriffenen Sichtweise des Symbolischen Interaktionismus (citeMead68). Von hier aus charakterisiert sie den explorativen Suchprozess als wiederholten Brückenbau zwischen dem Begriffsraum und dem Dokumentenraum (alle Arten von Informationsobjekten). Unter dem Begriffsraum wird dabei der »kognitive Bereich« (*cognitive realm*) verstanden, der benötigt wird, um Dokumente zu interpretieren. Hierzu gehören der Kontext und der momentane Wissensstand. Der Dokumentenraum ist die Sammlung aller Informationsobjekte.

Wie vollzieht sich nun dieser Brückenbau bei einer Recherche? Man betrachte zunächst die kompliziertere Richtung: vom Dokumentenraum zum Begriffsraum. Der Informationssuchende interpretiert die Ergebnisse im Dokumentenraum, die er zurückgeliefert bekommen hat. Er entdeckt bestimmte Muster und erweitert so sein begriffliches Verständnis. Gleichzeitig verschiebt er seine Aufmerksamkeit auf neue Begriffe, die er aus der Interpre-

tation gewonnen hat, und nimmt sie als Ausgangspunkt für seine weitere Recherche. Vom Begriffsraum in den Dokumentenraum gelangt er wieder, indem er sein Konzept in eine Suchaktion umsetzt. Interpretation und Suchaktion sind also die beiden Fahrspuren auf der Brücke zwischen Dokumentenraum und Begriffsraum.

Eine wichtige konzeptionelle Differenzierung von WANG BALDONADO ist die zwischen Begriffsraum und Dokumentenraum, zwischen denen sich der Suchende im Rahmen eines Suchprozesses ständig hin und her bewegen muss. Der Übergang vom Dokumenten- in den Begriffsraum ist wiederum ein Lernprozess, während die umgekehrte Richtung die Anwendung des individuellen Wissens darstellt.

Diese Betrachtung bestätigt – wie schon oben kurz angerissen –, dass es für den Lernprozess im Rahmen der Informationsrecherche von großer Bedeutung ist, dem Benutzer nicht einfach nur objektiv relevante Dokumente zu liefern, sondern dass diese auf der Basis des Vorwissens auch verstehbar sein müssen. Ansonsten funktioniert der Weg vom Dokumentenraum in den Begriffsraum nicht. Gleichzeitig zeigt das Modell aber auch die Grenzen für ein lernunterstützendes System auf: wesentliche Teile des Lernprozesses laufen allein im Kopf des Lernenden ab – hier kann ein lernunterstützendes System nur bedingt den aktuellen »Wissensstand« mitverfolgen.

5.4. Modelle für das Informationssuchverhalten

Der Informationssuchprozess als Modell für die Handlungen und die inneren Vorgänge bei einer Informationssuche ist ein etwas vernachlässigtes Forschungsgebiet. Ein klassisches Modell, das die unterschiedlichen Handlungen in einem Informationssuchprozess charakterisiert, stammt von ELLIS ([Ell89], [ECH93], [EH97]). Es war ursprünglich dazu gedacht, den Entwurf von Information-Retrieval-Systemen zu unterstützen, wurde später aber zu einem Analysemodell weiterentwickelt, das zur Beschreibung der Verhaltensweisen bei der Informationssuche im beruflichen Kontext dienen kann. Als grundlegende Handlungstypen oder -muster wurden die folgenden identifiziert:

- **Beginnen** (*starting*). Dies umfasst Aktivitäten in der initialen Phase der Informationssuche.
- **Verketteten** (*chaining*). Hierbei handelt es sich um das Verfolgen von Verweisen in Fußnoten oder im Literaturverzeichnis von vorliegenden Ressourcen, um zu neuen Ressourcen zu gelangen. Dies kann man auch auf das Hypermedia-Paradigma übertragen und hierunter das Verfolgen von Links subsumieren.
- **Stöbern** (*browsing*). Dies bezeichnet das überblicksartige Durchforsten eines gesamten Interessensgebietes nach Informationen.

- **Differenzieren** (*differentiating*). Hierbei wird versucht, die Informationen gegeneinander abzugrenzen und zu bewerten auf der Basis ihrer Natur und Qualität.
- **Überwachen** (*monitoring*). Um die Entwicklungen in einem bestimmten Themengebiet zu verfolgen, werden bestimmte Quellen überwacht.
- **Extrahieren** (*extracting*). Hierbei wird eine bestimmte Quelle systematisch durchgearbeitet, um interessantes Material aufzufinden.
- **Überprüfen** (*verifying*). Hierbei wird die Qualität der gefundenen Informationen gezielt überprüft.
- **Abschluss** (*ending*). Hierdurch wird der Rechercheprozess abgeschlossen.

Dabei wird keine Reihenfolge der Aktivitäten modelliert, sondern nur die vorkommenden Muster, da die Reihenfolge stark vom Kontext der konkreten Informationssuche abhängt.

BATES [Bat86] klassifiziert Suchaktivitäten nach zwei Dimensionen zielgerichtet vs. ungerichtet und aktiv vs. passiv in vier Aktivitätsklassen: Suchen (aktiv, zielgerichtet), stöbern (aktiv, ungerichtet), beobachten (zielgerichtet, passiv) und bewusst sein (ungerichtet, passiv).

Jüngere Modelle wie das von CHOO gehen speziell auf die Web-Suche ein [CDT00], bringen aber für diese Arbeit keine weiteren Erkenntnisse.

5.5. Just-in-Time Information Retrieval

Auch die Idee einer Informationsversorgung bei Bedarf wurde unter dem Begriff »just-in-time information retrieval« bereits untersucht. Hier ist speziell der Ansatz von RHODES zu nennen ([Rho00], [RM00]), der sog. JIT-IR-Agents untersucht. Ein solcher Agent sucht proaktiv nach relevanten Informationen und stellt sie so dar, dass sie leicht zugänglich für den Nutzer, aber immer noch so unauffällig präsentiert werden, dass sie die Arbeit des Nutzers nicht unterbrechen. Prototypisch wurden dabei drei Arten solcher Agenten umgesetzt: ein »Remembrance Agent« (zeigt in Beziehung zum aktuellen Dokument relevante Dokumente innerhalb eines Texteditors an), Margin Notes (integriert Links zu lokalen Dokumenten in aktuell angezeigte Web-Seiten) und Jimminy (ein tragbarer Rechner, der lokationsabhängige Informationen präsentiert).

Ein interessantes Ergebnis dieser Arbeit ist, dass die Nutzer durch die Reduktion der Suchbarrieren durch die Agenten zu einer erhöhten Nutzung von Informationen geführt werden. Zwar führen die automatisch generierten Anfragen zu geringerer Präzision, aber

dies wird durch den weggefallenen Aufwand zur Anfrageformulierung mehr als aufgewogen, wenn das Informationsbedürfnis nur latent vorhanden war. Was den Präsentationsmodus der Empfehlungen anbelangt, so ergab sich, dass sie dann am effizientesten sind, wenn sie möglichst verschieden zu der aktuellen Arbeitsumgebung sind. Weiterhin wurden gute Erfahrungen mit einer stufenweisen Tiefe der Informationspräsentation gesammelt, bei der der Benutzer den Grad der Ausführlichkeit selbst steuern kann.

5.6. Fazit

Mit dem oben skizzierten Wissensbegriff ist Wissen an sich unzugänglich für technische Systeme. Vielmehr müssen sie sich – um lernunterstützend zu wirken – auf das geeignete Zugänglichmachen von Informationen zurückziehen, die dazu geeignet sind, Wissen aufzubauen. Die Forschung um die Anatomie des Informationsrechercheprozesses hat wichtige Erkenntnisse beizusteuern. Informationssuche kann als schwach geführter Lernprozess verstanden werden, während die klassischen E-Learning-Lösungen von einer starken Führung und Vorstrukturierung ausgehen. Mit dem Konstruktivismus und seiner Betonung von offenen, problemorientierten Lernumgebungen verändert sich allerdings auch das Lernen stärker in Richtung Informationssuche/-recherche und Problemlösung. Hier – so hat sich gezeigt – existieren schon einige Ansätze wie z.B. KUHLETHAUS Modell der Informationsrecherche oder WANG BALDONADOS Ansatz zur Rechercheunterstützung. Allerdings scheint die Erkenntnis, dass nicht nur im konstruktivistisch verstandenen Lernen die Informationsrecherche einen wichtigen Bestandteil darstellt, sondern die Informationsrecherche selbst als Lernprozess verstanden werden muss, noch keinen vollständigen Niederschlag gefunden zu haben.

Dies mag auch daran liegen, dass die Informationsrechercheforschung besonders stark von den Bibliothekswissenschaften beeinflusst ist (und die meisten unterstützenden Studien wurden in diesem Kontext durchgeführt [JI04]). Die dort betrachtete Konzeption eines »Mediators«, der bei der Recherche helfend eingreifen kann, ist ohne weiteres mit einer Form von pädagogischen Führung im konstruktivistischen Sinne gleichzusetzen, so dass die Erkenntnisse zu Unterstützungsmöglichkeiten und -voraussetzungen sich übertragen lassen. NIEDZWIEDZKA und CHOO bieten eine gute Strukturierung, an welchen Stellen der Rechercheprozess durch kontextuelle Faktoren beeinflusst wird. Im einzelnen sind dies die Identifikation des Bedürfnisses, die Entscheidung zur Initiierung der Recherche/des Lernprozesses, die eigentliche Durchführungsphase, und die Auswahl/Verarbeitung des Gefundenen. Entlang dieser Punkte sollen die Schlussfolgerungen aus der Analyse in diesem Kapitel präsentiert werden:

- Für den Betroffenen ist bereits die Identifikation des Bedürfnisses mit einer hohen kognitiven Leistung verbunden. Ein latentes Bedürfnis muss nach seiner Bewusst-

werdung in die Sprache des Informationssystems übersetzt werden. Dies korrespondiert mit einer hohen kognitiven Belastung von außen (*extraneous cognitive load*) in the Cognitive-Load-Theorie von CHANDLER und SWELLER (vgl. [CS91], [Swe94]), die zusätzlich zur intrinsischen Belastung wirkt.

- Ob überhaupt eine Recherche initiiert wird, hängt neben der kognitiven Fähigkeit sehr stark von affektiven Faktoren ab. Hier übt die Selbsteinschätzung und das Selbstvertrauen generell und die Belastung der aktuellen Situation einen wesentlichen Einfluss aus. Auch die vergangene Erfahrungen mit Suchaktionen (oder Erfahrungen mit »selbstgesteuerten Lernprozessen«) wirken auf die Einschätzung, ob man den Prozess wohl erfolgreich abschließen wird – oder eben nicht. Alles spricht hier dafür, dass jegliches System sich hier sensibel auf die Persönlichkeitsstruktur des Lernenden und seine affektive Situation einstellen muss, um statt Frustration eine lernförderliche Grundstimmung erhalten oder sogar stimulieren zu können.
- Doch nicht nur die Initiierung des Prozesses ist ein komplexer Vorgang, sondern auch die gesamte Durchführung des Prozesses. Hier hat KUHLETHAU den wichtigen Erkenntnis den Weg gebahnt, dass die Informationsrecherche ein ganzheitlicher Lernprozess ist, der Denken, Fühlen und Handeln umfasst. Hier ist besonders das Phänomen der Unsicherheit des Suchenden zu nennen, die sich charakteristisch über die Phasen entwickelt. Mit der konstruktivistischen Grundannahme ist daraus zu schließen, dass sich diese Unsicherheit durch gezielte pädagogische Führung im akzeptablen Rahmen halten lässt. Dies schlägt sich im wesentlichen im Gleichgewicht von Redundanz und Einzigartigkeit nieder, das für Systeme bedeutet, dass sie in der Interaktion das Vorwissen des Benutzers berücksichtigen müssen und den Benutzer nicht überfordern dürfen, weil ansonsten evtl. Lernprozesse abgebrochen werden. Auch spricht vieles dafür, in den Anfangsphasen des Prozesses den Benutzer eher mit »reiferem« Material zu versorgen, das pädagogisch vorstrukturiert ist, bevor er sich in dem Themengebiet erfolgreich orientiert hat. Auch hier kann die Cognitive-Load-Theory eine Erklärung liefern: der hohe Grad an impliziter Kontextualisierung von unreifen Informationsartefakten erzeugt für Novizen aufgrund der fehlenden »Schemata« eine zu hohe kognitive Last, während die Redundanz von reifen Materialien die Last für Experten erhöht (vgl. auch [BO01]).
- WANG BALDONADO weist in ihrer Analyse der Einzelschritte darauf hin, dass die Welt des individuellen Wissens und die Welt der in den Rechercheprozessen genutzten Informationen voneinander getrennt zu betrachten sind, auch wenn gerade in Rechercheprozessen wiederholt dazwischen gewechselt wird. Dies stützt und präzisiert die Sichtweise auf die Reifungsschritte im Wissensreifungsprozess.
- WILSON schließlich hat systematisch auf der Basis von empirischen Ergebnissen die

Einflussfaktoren klassifiziert. Dies wird die Grundlage für die Modellierung des Kontextes liefern.

Vor einer Identifikation der Informationsrecherche mit E-Learning oder Lernen an sich sei allerdings gewarnt. Die Informationsrechercheforschung realisiert zwar – besonders in ihrer konstruktivistischen Ausprägung – die komplexen Lernprozesse, jedoch bleibt der Blickwinkel einseitiger. Es fehlen wesentliche Aspekte wie z.B. die Fragen nach definierten Lernzielen, die soziale Dimension des Lernens, die Problematik der Lernerfolgskontrolle u.v.m. Letztendlich sollte die Schlussfolgerung dieses Kapitels auch sein, dass die Informationsrecherche als *ein* Lernprozess aufgefasst werden sollte, der aus einer Sammlung von unterschiedlichen Lernformen besteht, und die Informationsrecherche ein wichtiger Bestandteil von offenen, wenig geführten Lernprozessen sein kann. Für den Wissensreifungsprozess spielt die Informationsrecherche in allen Phasen des Reifungsprozesses eine Rolle, allerdings ist zu berücksichtigen, dass der Reifegrad – wie oben ausgeführt – durchaus einen Unterschied macht. Diese Reifungsperspektive kann also auch die Informationsrechercheforschung um eine zusätzliche Dimension bereichern, die nicht nur die Führung durch Mediatoren, sondern auch durch die Informationen und ihre Präsentation selbst berücksichtigt.

Abschließend sei angemerkt, dass während sich die Beziehung von Informationsrecherche und Lernprozessen recht gut herausarbeiten lässt, erstaunlicherweise die Beziehung zu Arbeitsprozessen und -aufgaben in der bisherigen Forschung fast vollständig ausgeblendet worden ist [IJ05].

6.

Methodik des situationsgesteuerten Lernens

Nach der Analyse der Beiträge der Informationsrecherche Forschung tritt klarer hervor, wie wir zwischen dem kursorientierten Paradigma und dem der reinen selbstgesteuerten Suche das situationsgesteuerte Lernen positionieren können. Dabei lassen sich auf beiden Seiten aktuelle Entwicklungen als konvergent identifizieren. Zum einen zeigen konstruktivistisch motivierte Informationsrechercheprozessmodelle, dass man auch Suche als Lernprozess verstehen muss, der der Führung bedarf und liefert hierzu Interventionspunkte. Zum anderen entwickelt sich E-Learning (insbesondere unter dem Etikett »E-Learning 2.0«) weg von einer Fremdsteuerung hin zu einer stärkeren Selbststeuerung der Lernprozesse, in denen die Aktivität vom Lernenden ausgeht. Hierbei wird jedoch eine Vorstrukturierung und »Vor-Gestaltung« nicht aufgegeben. Dem situationsgesteuerten Lernen fällt hierbei die Aufgabe zu, die beiden Ansätze zu verbinden (vgl. Abb. 6.1).

In diesem Kapitel soll vor dem Hintergrund der bisher herausgearbeiteten Eigenschaften des arbeitsbegleitenden Lernen und der Ergebnisse aus dem vorangegangenen Kapitel über die Informationsrecherche die Methodik des bislang informell eingeführten »situationsgesteuerten Lernens« dargestellt werden. Die Bezeichnung »situationsgesteuertes Lernen« wurde deshalb gewählt, weil die Situation das zentrale Element ist, die Lernziele und mögliche Inhalte, Zeitpunkte und Interventionen definiert. Grundlage hierfür ist das Paradigma der Lernunterstützung, indem davon ausgegangen wird, dass die Entscheidungsgewalt über das Einleiten und Verfolgen von Lernaktivitäten beim Lernenden liegt, der aber in seiner Entscheidung unterstützt wird durch Angebote durch das lernunterstützende System.

6.1. Grundidee

Die Grundidee des situationsgesteuerten Lernens ist, dass der Mitarbeiter seinem Alltagsgeschäft nachgeht, ohne dass er sich ausdrücklich weiterqualifizieren möchte. Das System

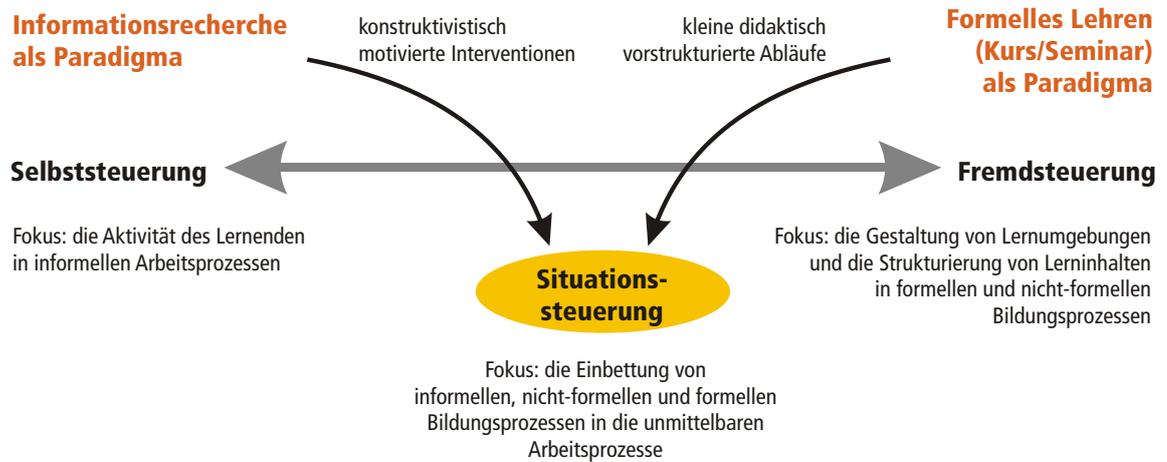


Abbildung 6.1.: Situationssteuerung zwischen den Polen Fremd- und Selbststeuerung

verfolgt die Aktionen des Benutzers. Auf der Basis von Veränderungen und der aktuellen Wissenslücke kann das System dem Lernenden signalisieren, dass bestimmte Lernmöglichkeiten existieren. Dies geschieht beispielsweise durch periphere Anzeigetechniken oder durch integrierte Hinweise in der Benutzerschnittstelle seiner Anwendungen. Der Benutzer kann entscheiden, ob er eine der Empfehlungen annimmt und mit dem Lernen beginnen will, oder die Empfehlung auf später verschiebt oder ganz ignoriert. Im ersten Fall stellt das System ein Lernprogramm auf der Basis der Empfehlung und der Benutzerauswahl zusammen, die auf den Benutzer abgestimmt ist, also insbesondere auf sein Vorwissen, seine technischen Gegebenheiten und seine Präferenzen. Mit anderen Worten meint »situationsgesteuertes Lernen«, dass die Situation eines Mitarbeiters und ihre Änderung über die Zeit hinweg das Wann, Was und Wie des Lernens maßgeblich beeinflusst, wobei dem Lernenden stets die Möglichkeit offensteht, in ein selbstgesteuertes Lernen überzugehen. Hierbei wird der Begriff der **Situation** (in Anlehnung an [MH68], [Sch02b] oder [Has05]) zeitabhängig (und somit über die Zeit hinweg veränderlich) verstanden:

Definition 6.1 (Situation) Eine Situation ist der Zustand der realen Welt zu einem bestimmten Zeitpunkt.

Definition 6.2 (Benutzungssituation) Eine Benutzungssituation (auch: Nutzungssituation, Benutzersituation) ist jener Ausschnitt der Situation zum Nutzungszeitpunkt, der einen Einfluss auf die Benutzung eines Systems durch einen bestimmten Benutzer hat.

Im Gegensatz zur globalen Situation findet hier eine Einschränkung der Perspektive auf eine einzelne Person statt, die in ihrer Rolle als Benutzer mit einem System interagiert.

Hierfür ist nur ein bestimmter Ausschnitt relevant. Dies entspricht der Begriffsbildung *context of interaction* in [CR02].

Diese Lernform des situationsgesteuerten Lernens hat besonders gegenüber den beiden anderen Lernformen, die heute von Lernplattformen unterstützt werden, folgende Vorteile:

- *Gegenüber kursgesteuertem Lernen.* Durch die feinere Granularität ermöglicht das situationsgesteuerte Lernen eine bessere Abstimmung mit dem aktuellen Wissensbedürfnis. Damit ist es nicht nur für strategisches Lernen, sondern insbesondere auch für operatives Lernen geeignet.
- *Gegenüber selbstgesteuertem Lernen.* Es ist zu beachten, daß sowohl Initiierung als auch Steuerung eines Lernprozesses kognitiv anspruchsvoll ist und deshalb nicht für jeden Mitarbeiter, aber auch nicht für jede Arbeitssituation geeignet ist, da es die Bewußtmachung des Unsicherheitsgefühls und dessen Übersetzung in eine Problemformulierung erfordert (vgl. Informationssuchprozess [Kuh04]). Durch die Unterstützung bei Initiierung und Steuerung reduziert hier das situationsgesteuerte Lernen die kognitive Last und hilft so, affektive Barrieren gegenüber Lernaktivitäten abzumildern.

Im situationsgesteuerten Lernen wird **Lernunterstützung** durch die Nutzung von Informationsdiensten in drei Formen umgesetzt:

- als *Intervention* in individuelle Lernprozesse, was basierend auf den Erkenntnissen zur Informationssuche primär als Initiierung verstanden wird,
- als *Strukturierung* im vereinfachten Sinne eines Scaffolding, das ein gewisses Grundgerüst liefert durch automatische Auswahl, Ordnung und Empfehlung von Lerngelegenheiten sowie
- als *Umgebungsgestaltung*, die Lerngelegenheiten stärker mit der aktuellen Arbeitssituation in Beziehung setzt.

6.2. Ablauf anhand der Beispielszenarien

Bevor das abstrakte Prozessmodell vorgestellt wird, sollen in einem ersten Schritt die in Abschnitt 4.2 eingeführten Szenarien konkretisiert und so veranschaulicht werden.

6.2.1. Szenario 1: Kundenberater im Logistikvertrieb

Der Kundenberater meldet sich im System an und beginnt seine Arbeit. Üblicherweise ist er speziell für Massenversandprodukte zuständig. Da sein Kollege heute krank ist, ist er auch noch für Anfragen bzgl. Expresssendungen von Großkunden zuständig. Das System informiert ihn, dass sich die Produktbedingungen für den Massenversand (Bulk-Mail) ins Ausland verändert haben und empfiehlt ihm ein Lernobjekt, das die ihm die geänderten Regelungen und ihre Folgen für den Kunden und die Beratung illustriert. Weiterhin werden ihm ein paar Lernobjekte zu den Express-Produkten angezeigt. Der Mitarbeiter entscheidet sich für das Aktualisierungslernobjekt und arbeitet dieses durch. Zur Sicherung der Lernkontrolle ist am Ende des Lernobjektes ein kleiner Wissenstest, den er ohne Fehler besteht. Dies wird in seinem Portfolio vermerkt.

Er beginnt mit seiner Arbeit. Als er eine telefonische Anfrage für ein schriftliches Angebot für eine größere Anzahl von Expressendungen erhält, merkt er, dass er sich im Arbeitsbereich seines Kollegen nicht mehr so gut auskennt. Er erinnert sich an die Empfehlungen des Lernsystems und ruft sie nochmals auf. Hier werden ihm spezielle Lernobjekte für Expressprodukte angeboten. Als er eines davon auswählt, wird für ihn ein Lernprogramm gestartet, das ihm die Grundlagen für die Expressprodukte vermittelt und dann ein spezielles Produkt vertieft. Anschließend wählt er das zweite empfohlene Objekt aus, das eine Simulation des Vertriebsprogrammes enthält. Die Simulation erlaubt ihm, den Prozess der Angebotserfassung anhand realitätsnaher Daten, die direkt übernommen werden, durchzuspielen, ohne dass reale Prozesse angestoßen werden. Hierbei werden ihm keine Grundlagen mehr angezeigt, da er diese bereits erfolgreich durchgearbeitet hat. Alle Lernaktivitäten werden im Portfolio vermerkt. Und er kehrt in seinen Arbeitsprozess zurück, um dort das angeforderte Angebot anzufertigen. Da stellt er fest, dass er sich doch noch nicht so vertraut mit der Thematik fühlt. Er ruft sein Portfolio auf, um das kürzlich bearbeitete Lernobjekt aufzurufen. Dort werden als weiterführende Materialien u.a. Musterangebote angezeigt. Er ruft eines davon auf und kann damit die Aufgabe gut bewältigen. Er vermerkt im Portfolio als Notiz das, was ihm noch gefehlt hatte.

6.2.2. Szenario 2: Projektmanagement

Die Projektmanagerin ist für mehrere Projekte zuständig. Als sie im Projektmanagementportal in ein EU-Projekt wechselt und dort die ihr als nächsten Prozessschritt angezeigte Aufgabe »Management Progress Report« auswählt, werden ihr in einer Seitenleiste des Portals diverse Dokumente zum Thema angezeigt, allerdings kein Lernobjekt, da hierfür bislang noch keines erstellt wurde. Insbesondere werden ihr auch noch Personen in ihrem Unternehmen angezeigt, die diese Aufgabe kürzlich abgeschlossen haben. Da sie weiß, dass sich etwas in der Vorgehensweise geändert hat, nimmt sie Kontakt mit einer der angezeigten Per-

sonen auf. Da sie in der Vergangenheit asynchrone Kommunikationswege bevorzugt hat, öffnet sich automatisch ein Mailfenster, über das sie ihre Frage an die empfohlene Person schicken kann. Diese antwortet eine Stunde später – in der Zwischenzeit hat sie sich anderen Aufgaben gewidmet. Sie trägt die Information als Notiz in ihr Portfolio ein und beginnt nun damit, ihre ursprüngliche Aufgabe zu bearbeiten. Dabei benutzt sie auch Vorlagen, die ihr vom System vorgeschlagen werden und aus vergangenen Projekten stammen.

6.2.3. Szenario 3: Softwareentwicklung

Vom Kunden wurde über das Ticketsystem ein Fehler gemeldet. Um diesen zu beheben, öffnet der Entwickler das entsprechende Projekt und wechselt in den Quelltext der Komponente, in der er den Fehler vermutet. Die Komponente benutzt die OWL-API zum Zugriff auf OWL-Ontologien. Leider hat er weder die entsprechende API noch Ontologien bislang benutzt, so dass er Schwierigkeiten hat, den Quelltext zu verstehen. Glücklicherweise existiert ein Kurztutorial zur API, das ein Kollege vor kurzem erstellt hat und das ihm vom System empfohlen wird. Als er dieses auswählt, wird ihm automatisch ein kleiner Kurs präsentiert, der ihm eine Kurzeinführung in Ontologien und speziell OWL gibt, bevor ihm das Schritt-für-Schritt-Tutorial für die OWL-API angezeigt wird, das er auch in einer Testumgebung ausprobiert. Er fügt ein Lesezeichen zu diesem Tutorial seinem Portfolio hinzu. Da er dem System auch mitgeteilt hat, dass er sich im Bereich semantische Technologien weiterqualifizieren möchte, wird ihm zusätzlich ein in zwei Wochen stattfindendes Präsenzseminar empfohlen, für das er sich auch gleich anmeldet.

Dann beginnt er mit der Fehlersuche in der Komponente. Bei bestimmten Aufrufen der API stellt er fest, dass diese sich nicht so verhalten, wie er erwartet. Das System bietet ihm Diskussionsforen an, in denen er recherchieren und ggf. selbst Fragen stellen kann.

6.3. Abstraktes Modell

Wie lassen sich die vorausgegangenen Beispiele in ein allgemeingültiges Modell abstrahieren? Für ein Modell, das sich auch umsetzen lässt, ist dabei eine ausgewogene Perspektive erforderlich, die die Sicht des Lernenden mit der Systemsicht verbindet. Deshalb soll im folgenden zunächst die Sicht des Lernenden und danach die Sicht des Systems betrachtet werden, die in Abb. 6.2 zusammengeführt werden.

6.3.1. Sicht des Lernenden

Ausgangspunkt aus Sicht des Lernenden ist im situationsgesteuerten Lernen stets der Arbeitsprozess (*Arbeiten*, vgl. Abb. 6.2; die rechteckigen Symbole stehen für die vom Lernenden sichtbaren Systemteile). Die jeweilige Arbeitssituation wird durch das lernunterstüt-

zende System mitverfolgt, und auf dieser Basis kann es *Empfehlungen* für Lerngelegenheiten präsentieren, die für die jeweilige Situation benötigte Kompetenzen vermitteln können. Diese Empfehlungen werden dem Lernenden i.d.R. so präsentiert, dass er die Wahl hat, ob er seinen Arbeitsprozess unterbrechen möchte oder nicht. Dies geschieht in Form einer Präsentation, die nur eine periphere Aufmerksamkeit des Lernenden erfordert.

Entscheidet sich der Lernende für die Empfehlungen, so wird je nach empfohlener Ressource entweder die *Lernumgebung* (Lernobjekt oder -programm oder das selektierte Dokument¹) oder eine oder mehrere Kommunikationsmöglichkeiten (mit einer *Lerngruppe/Community* oder einem *Tutoren/Experten*) geöffnet. Dabei ist es durchaus möglich, auch aus der Lernumgebung heraus mit anderen zu kommunizieren oder auf unterstützende Dokumente zuzugreifen. Auch der Zugriff auf weitere Lernobjekte in Form eines explorativen Lernens ist vorgesehen, in den das System nur noch rein reaktiv eingreift. Zum Abschluss eines Lernprozesses (bzw. eines Abschnitts) kann eine Überprüfung des Lernerfolgs (*Erfolgskontrolle*) im Sinne eines Tests stattfinden, über den versucht wird zu messen, ob das Lernziel erreicht wurde. Die Lernaktivitäten und -ergebnisse werden in einem *Portfolio* dokumentiert, so dass zum einen das System seine weiteren Empfehlungen daran anpassen kann, zum anderen der Lernende einen Überblick über seine Lernaktivitäten hat und ggf. gezielt auf bestimmte Lernressourcen zurückgreifen kann.

Anschließend kehrt der Lernende in seinen Arbeitsprozess zurück, wo die erworbene Kompetenz sofort angewandt werden kann.

6.3.2. Sicht des Systems

Die Rolle des Systems (vgl. die abgerundeten Symbole in Abb. 6.2) lässt sich mit Führung oder Steuerung des Lernprozesses umschreiben. Diese Führung reicht vom Beginn des Lernprozesses bis zum Abschluss, wobei die Intensität der Führung im Vergleich zum kursgesteuerten Lernen deutlich reduziert ist, da dem Lernenden mehr Spielräume eingeräumt werden. Diese Spielräume sind im Vergleich zum selbstgesteuerten Lernen kaum reduziert, da die Steuerung im wesentlichen als Unterstützung konzipiert ist.

Im folgenden sollen die einzelnen Elemente dieser Unterstützung näher analysiert werden. Diese orientieren sich am oben dargestellten Verlauf des situationsgesteuerten Lernens:

- **Initiieren.** Zwischen dem Arbeitsprozess und der Präsentation von Empfehlungen steht die Initiierungsphase, in der das System beurteilt, ob ein Lernprozess sinnvoll zu beginnen ist und was, wann und wie als Lerngelegenheit zu empfehlen ist. Hierbei wird also auf *Intervention* gesetzt.

¹Der Vereinfachung wegen wird konzeptionell davon ausgegangen, dass auch einfache Dokumente in einer Lernumgebung angezeigt werden – ohne dass diese freilich von den erweiterten Funktionen wie Lernfortschrittskontrolle o.ä. profitieren können

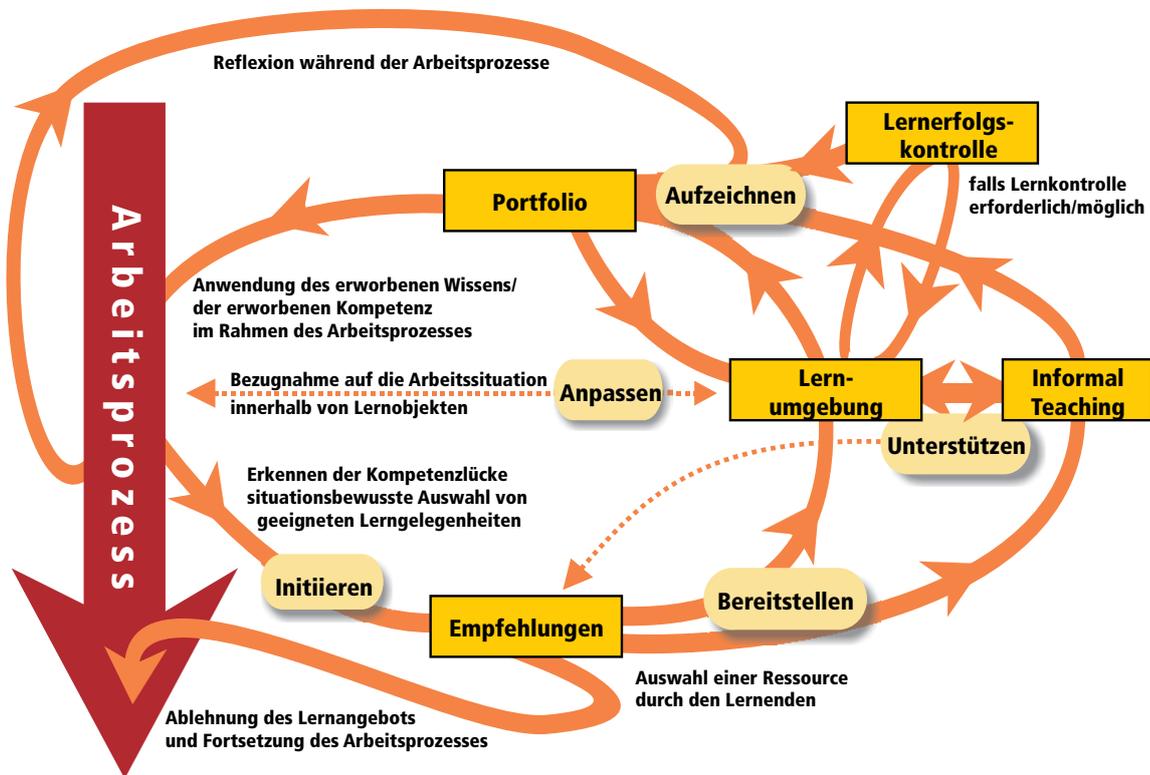


Abbildung 6.2.: Situationsgesteuertes Lernen: Prozessübersicht

- **Bereitstellen.** Hat sich der Mitarbeiter entschlossen, der Empfehlung zu folgen, geht es darum, die Lerngelegenheit angemessen bereitzustellen. Bei Lernobjekten ist darauf zu achten, dass der Lernende auch mutmaßlich in der Lage ist, diese zu verstehen. Bei Kollegen oder Experten ist der Kommunikationsweg zu beachten, der in der Situation am angemessensten ist. Dies entspricht im wesentlichen einer *Strukturierung*.
- **Unterstützen.** Wird die Lerngelegenheit genutzt, so muss sich das System im wesentlichen zurückhalten. Im Gegensatz zu behavioristischen Ansätzen wird bei konstruktivistischer Grundhaltung davon ausgegangen, dass man dem Lernenden eine offene Umgebung bietet, innerhalb derer er das Lernen selbst organisiert. Das System muss versuchen, sich auf *Umgebungsgestaltung* zurückzuziehen. Hierzu dient zum einen der Übergang zu anderen Arten von Lerngelegenheiten (Lerngruppen in Form von Diskussionsforen o.ä.), zum anderen das Angebot von Navigationsmöglichkeiten.
- **Aufzeichnen.** Um informelle Lernaktivitäten sowohl für die Reflexion als auch für

die externe Beurteilung zu dokumentieren, werden Lernaktivitäten samt ihren Ergebnissen und persönlichen Notizen gespeichert. Reflexion kann auch erst im Rahmen der Anwendung bzw. im kritischen Rückblick darauf stattfinden. Die Möglichkeit zur Aufzeichnung kann dabei ebenfalls als ein Element der *Umgebungsgestaltung* verstanden werden.

Im folgenden werden diese Elemente (Initiieren, Bereitstellen, Unterstützen, Aufzeichnen) noch weiter untergliedert.

Initiieren

Wie im Kapitel über die Informationsrecherche herausgearbeitet, liegt einer der kritischen Punkte bereits bei der Initiierung von Such- bzw. Lernprozessen. Der Übergang zwischen Arbeits- und Lernprozessen ist kognitiv anspruchsvoll. Dementsprechend ist hier das System gefordert, proaktiv zu handeln, anstatt wie klassische Lernplattformen rein reaktiv. Dieses proaktive Verhalten lässt sich in die folgenden Schritte/ Aspekte zerlegen:

- *Erkennen der Lernmöglichkeit.* Das System muss die Arbeitsprozesse des Mitarbeiters verfolgen. Ausgelöst durch Veränderungen in der Arbeitssituation muss das System versuchen zu ermitteln, ob ein Übergang in Lernprozesse notwendig oder sinnvoll ist. Diese Systementscheidung sollte sich an den Anforderungen an die Kompetenz des Mitarbeiters in der jeweiligen Situation orientieren.
- *Ermitteln der Empfehlungen (was).* In diesem Schritt bestimmt das System, was es dem Mitarbeiter als mögliche Lerngelegenheit anbieten kann. Hierzu werden relevante Lernobjekte, Dokumente oder Personen bestimmt und entsprechend ihrer (mutmaßlichen) Relevanz für die aktuelle Situation geordnet.
- *Wählen des Zeitpunktes der Präsentation der Empfehlungen (wann).* Sind die Empfehlungen ermittelt, geht es um die Frage, wann man die Empfehlungen dem Mitarbeiter präsentiert. Hierbei muss vermieden werden, den Benutzer durch nicht situationsadäquates Verhalten zu verärgern, weil er aus seiner Konzentration auf die Arbeitsaufgabe gerissen wird, ohne dass die Empfehlungen für ihn einen hinreichenden Nutzen haben. Das System muss also evtl. ermittelte Empfehlungen auch zurückhalten, bis z.B. der Mitarbeiter »Zeit« hat oder seine affektive Grundeinstellung offen gegenüber dem Lernen ist².
- *Wählen der Art der Präsentation der Empfehlungen (wie).* Verknüpft mit der Frage nach dem Zeitpunkt der Empfehlung ist die Frage nach dem Wie. Hierbei geht es im wesentlichen um die Aufmerksamkeit, die dem Mitarbeiter abverlangt wird. Die Art

²Dies lässt sich beispielsweise mit Hilfe von Ansätzen zur Ermittlung von Frustration wie z.B. [Rey01] ermitteln.

der Präsentation kann dabei von einem Erzwingen des Übergangs in einen Lernprozess bishin zu Systemmeldungen im peripheren Wahrnehmungsbereich reichen, wodurch man unerwünschte Störungen vermeidet. Eine Entscheidung für eine bestimmte Art der Präsentation kann hierbei von unterschiedlichen Faktoren abhängig gemacht werden, z.B. wie wichtig es vom Unternehmen eingestuft wird, wie sicher sich das System ist, dass der ermittelte Kontext der realen Situation entspricht o.ä.

Bereitstellen

Die Rolle des Systems ist in der Bereitstellungsphase zu unterscheiden nach der Art der Lerngelegenheit: explizite Lernressource (speziell bei Lernobjekten) oder »Mensch«:

- *Generierung von Lernprogrammen aus Lernobjekten (Berücksichtigung von Abhängigkeiten).* Im Unterschied zu traditionellen Information-Retrieval-Systemen (und die auf ihnen aufsetzenden Wissensmanagementsysteme) reicht es selbst bei in sich abgeschlossenen Lernobjekten oft nicht aus, einfach nur dieses Objekt dem Lernenden zu präsentieren, weil der Lernende nicht in der Lage ist, das Objekt zu »verstehen«, weil relevantes Vorwissen oder auch nur die Brücke zur aktuellen Situation fehlt, die den Lernenden dort abholt, wo er sich gerade befindet. Durch eine Zusammenstellung eines Lernprogrammes, das alle Voraussetzungen vermittelt, lässt sich die Unsicherheit reduzieren, die, wie im Modell von KUHLLTHAU dargelegt, ein Informationsbedürfnis begleitet. Gerade diese Unsicherheit wird durch zuviel »Neues« noch verstärkt.
- *Wählen der Kommunikationsform und des Kommunikationszeitpunktes für Personen.* Gerade bei den informelleren Lernprozessen steht die Kommunikation im Vordergrund. Auch hier ist es von der Situation abhängig, welche Kommunikationsform die geeignete ist – und zwar nicht nur hinsichtlich der Situation des Lernenden, sondern auch der Situation des Experten bzw. Tutors. So kann das System hier zwischen synchronen und asynchronen Kommunikationsformen wählen.

Unterstützen und Anpassen

Sind die Lerngelegenheiten bereitgestellt, so ist auch hier das System gefragt, den Lernenden in der Interaktion zu unterstützen. Hierzu gehört:

- *Adaption der Präsentation der Lerninhalte.* Hierbei wird die Darstellung der Inhalte an den Lernenden und seine technische Umgebung angepasst, was vor allem im Bereich des mobilen Lernens (also Lernen mit mobilen Geräten) von Bedeutung ist, wo die Möglichkeiten der Geräte sehr beschränkt ist. Aber auch auf der Ebene der Lernerpräferenzen (z.B. auf der Basis einer Lernertypologie) kann eine Anpassung Sinn

machen (z.B. knappe Darstellung vs. ausführliche Erläuterungen mit illustrierenden Bildern).

- *Verhaltensadaption der Lernobjekte.* Aus didaktischer Sicht ist ein hoher Realitätsbezug lernförderlich, so dass Lernobjekte auf die Arbeitssituation (die den späteren Anwendungskontext darstellt) eingehen sollten. So könnte man z.B. in einem Beispiel im Lernobjekt auf die aktuelle Arbeitssituation Bezug nehmen. Dies kann allerdings nur dadurch erfolgen, dass (a) eine gewisse Adaptionlogik in den Lernobjekten selbst enthalten ist und (b) diese Logik in die Lage versetzt wird, an Informationen über die Situation zu gelangen.
- *Generieren von Navigationsmöglichkeiten.* Wenn man dem Lernenden ein gewisses Maß an Selbststeuerung geben will, so ist es unerlässlich, daß die Lernumgebung in dem Sinne offen ist, dass sie den Zugriff auf beliebige Ressourcen ermöglicht – nicht nur auf die, die vom System als relevant eingestuft wurden. Allerdings ist auch hier eine Unterstützung sinnvoll, die gezielte Navigationsmöglichkeiten anbietet zu Lernobjekten, die formell oder informell verbunden sind.

Aufzeichnen

Das Aufzeichnen und Dokumentieren der Ergebnisse von Lernaktivitäten ist in der Praxis von großer Bedeutung und hat (wie schon in Abschnitt 2.5 im Rahmen der elektronischen Portfolios skizziert) viele unterschiedliche Facetten. Hierzu gehören:

- *Dokumentation von Lernaktivitäten.* Hiermit kann festgehalten werden, wie sich ein Mitarbeiter engagiert hat, um seine derzeitigen Kompetenzen auszubauen. Hierdurch wird ein analoger Mechanismus zu den Teilnahmebescheinigungen bei Präsenzseminaren geschaffen, der es erlaubt Lernaktivitäten nachzuvollziehen. Wie diese genutzt werden, ist dabei offen, wenn auch der Umgang damit durchaus ein sensibles Feld ist.
- *Dokumentation des Lernerfolgs.* Hierzu können die Ergebnisse von formalisierten Tests festgehalten werden, wie sie vor allem Bestandteil von großen pädagogisch aufbereiteten Kursen sind. Dies ist beispielsweise im Rahmen von Zertifizierungen wichtig, wo eine bestimmte Anzahl an korrekten Antworten Bestandteil der Anforderungen an die Zertifikatsvergabe ist.
- *Reflexion.* Die in informellen Lernprozessen vielleicht wichtigste, gleichzeitig aber auch derzeit noch unzureichend erforschte Funktion ist die Reflexion des Mitarbeiters über seine Arbeits- und Lernprozesse und die dort gemachten Erfahrungen. Diese findet meist nicht im direkten Anschluss an die Lernprozesse statt, sondern davon

abgekoppelt. Ein wichtiger Bestandteil ist dabei die soziale Dimension, so dass hierfür vermehrt öffentliche Blogs als Portfoliobestandteil vorgeschlagen werden.

6.4. Fazit

In diesem Kapitel wurde die Lernunterstützungsmethode des situationsgesteuerten Lernens vorgestellt, die die Grundlage und das Anwendungsszenario der in den folgenden Kapitel auszuarbeitenden technischen Lösung darstellt. Diese Methode wurde auf der Basis der Erkenntnisse aus den vorangegangenen Kapitel entwickelt, was in Abb. 6.3 nochmals systematisch aufbereitet ist:

- Die Problemorientierung in konstruktivistischen Ansätzen (insbesondere im situier-ten Lernen) wurde mit der Arbeitssituation gleichgesetzt, wobei der Fokus auf dem *Erkennen und Erhalten* dieser Situation liegt.
- Auf der Grundlage des Modells für konstruktivistische Lernumgebungen (als Mi-kromodell für das arbeitsbegleitende Lernen) (Abschnitt 2.8.1) sowie vor dem Hin-tergrund des Wissensreifungsprozesses wurde das Konzept der Lerngelegenheit ent-wickelt, das mehr als nur klassische Lernressourcen umfasst und auch informelle Lerngelegenheiten mit einschließt.
- Die Kombination aus Scaffolding und den Modellen zur Informationsrecherche bil-dete die Basis für die Charakterisierung der Form der Führung im situationgesteu-erten Lernen, die durch die Formen Intervention, Strukturierung und Umgebungsge-staltung operationalisiert wird, wobei vor allem die Intervention sich aus der Infor-mationsrecherche ableitet.
- Die systemgestützte Lernunterstützung als Realisierung der drei Führungsformen wurde in einzelne »Systemprimitive« zerlegt (Initiieren, Bereitstellen, Unterstützen, Anpassen und Aufzeichnen), die jeweils auf die Situation zurückgreifen.
- Ein wichtiges Konzept aus der Analyse der Informationsrecherche ist die *subjektive Relevanz*, die die Umsetzung der beiden Systemfunktionen *Initiieren* und *Bereitstellen* maßgeblich beeinflusst.

Im folgenden werden aus dieser Zerlegung der Lernunterstützung die Anforderungen für die lernunterstützenden Dienste abgeleitet, die Thema des dritten Teiles sind.

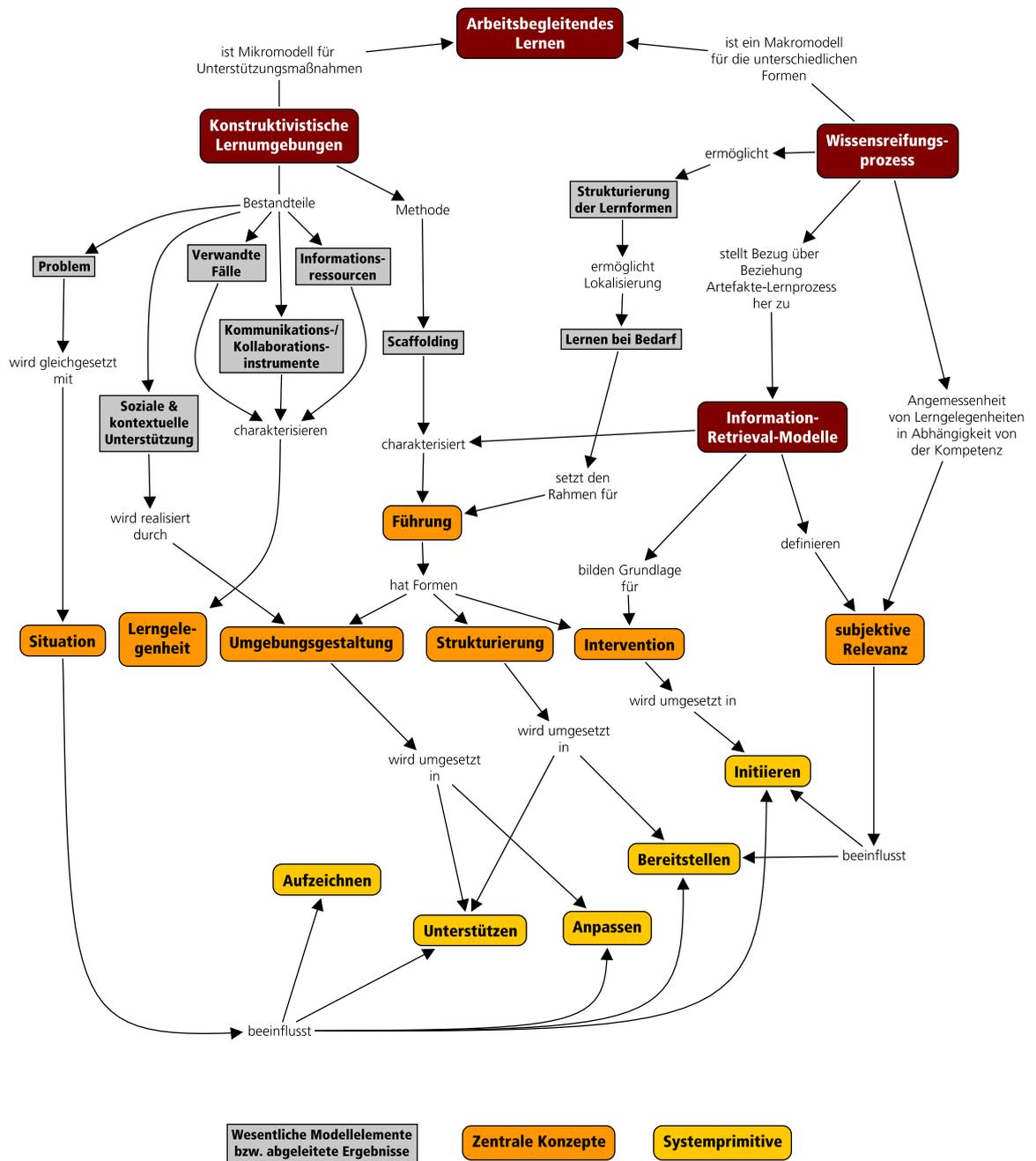


Abbildung 6.3.: Situationsgesteuertes Lernen im Kontext

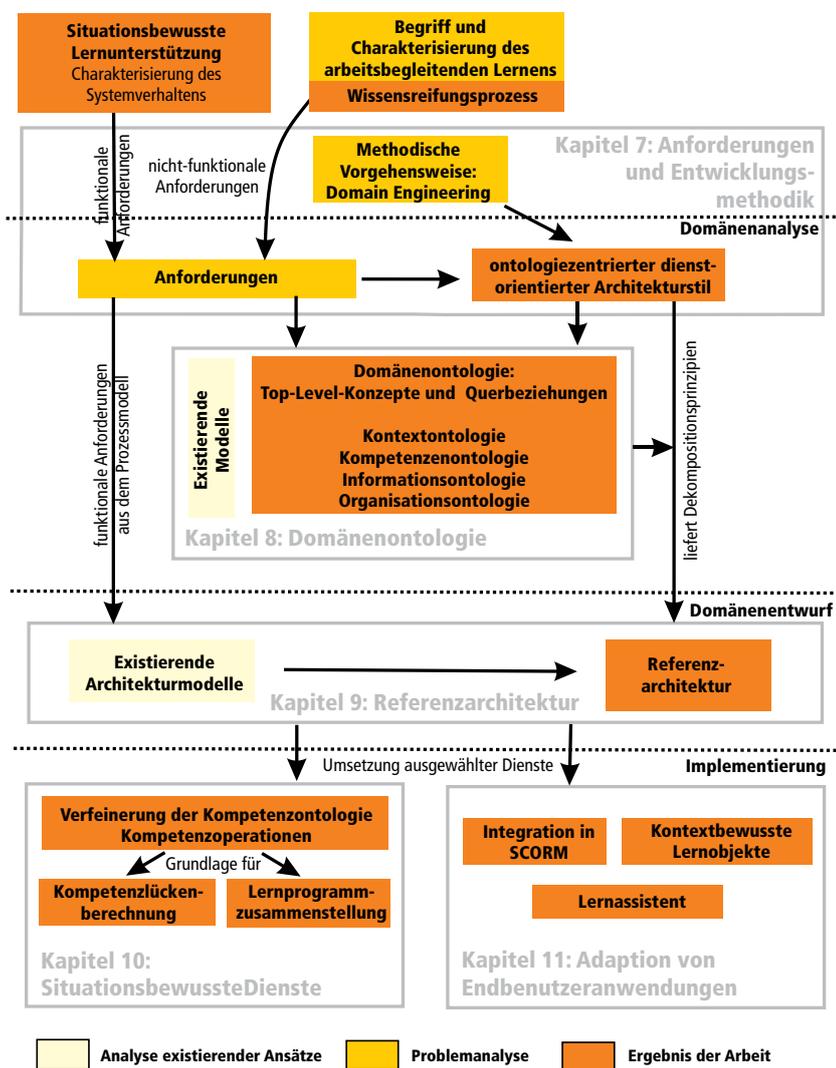
Teil III.

Dienstinfrastruktur für das situationsgesteuerte Lernen

Problemstellung

Für die technische Umsetzung des situationsgesteuerten Lernens wird eine Infrastruktur benötigt. Für diese müssen aus den Analysen der vorangegangenen Kapitel Anforderungen ermittelt und anschließend ein Rahmen konzipiert werden, der wiederverwendbare Dienste bereitstellt, die von lernunterstützenden Anwendungen benötigt werden.

Vorgehensweise



7.

Entwicklungsmethodik und Anforderungen für ein Dienstrahmenwerk

Nachdem im vorangegangenen Kapitel eine Methodik erarbeitet wurde (das situationsgesteuerte Lernen), wie sich das Lernen bei Bedarf vor dem Hintergrund des Wissensreifeungsprozesses informationstechnisch unterstützen lässt, geht es nun in diesem Teil darum, diese Unterstützung technisch zu konzipieren und exemplarisch umzusetzen.

In einem ersten Schritt geht es in diesem Kapitel um die Identifikation von Anforderungen, bevor in einem zweiten Schritt die weitere Vorgehensweise erarbeitet wird.

7.1. Funktionale Anforderungen

7.1.1. Funktionale Anforderungen aus der Beschreibung des Systemverhaltens

Aus dem Prozessmodell zum situationsgesteuerten Lernen in Abschnitt 6.3.2 in Kombination mit dem Referenzmodell für konstruktivistische Lernumgebungen (Abschnitt 2.8.1) lassen sich bereits die wesentlichen funktionalen Anforderungen extrahieren:

- **Initiieren.**

AF1 Das System muss automatisiert aus der Situation des Lernenden einen Lernbedarf erkennen.

AF2 Das System muss automatisiert feststellen können, ob für diesen Lernbedarf und die Situation relevante Lerngelegenheiten dem System bekannt sind.

AF3 Das System muss den Lernenden proaktiv über existierende Lerngelegenheiten informieren und ihm für die Situation relevante Lerngelegenheiten zur Auswahl präsentieren. Hierbei sind Zeitpunkt und Art der Präsentation auf ihre An-

gemessenheit hinsichtlich der Situation des Lernenden abzustimmen, so dass negative Effekte der Unterbrechung der Arbeitsprozesse weitgehend vermieden werden.

- **Bereitstellen.**

AF4 Bei der Bereitstellung von pädagogisch aufbereitetem Material muss berücksichtigt werden, ob der Lernende mit seinem derzeitigen Kenntnisstand diese auch verstehen kann, damit sie für ihn subjektiv relevant sind; soweit dies möglich ist, sollte auch für nicht pädagogisch aufbereitetes Material die subjektive Relevanz berücksichtigt werden (vgl. Abschnitt 5.1.4).

AF5 Bei der Kontaktaufnahme mit Kollegen muss auch deren Situation berücksichtigt werden, um auch hier störende Unterbrechungen soweit wie möglich zu vermeiden.

AF6 Existierende Kommunikationsanwendungen (wie EMail, Instant Messaging o.ä.) sollen nicht ersetzt, sondern nur angepasst werden.

- **Unterstützen.**

AF6 Zur besseren Einpassung in die Situation des Lernenden sollen Lernobjekte vom System mit Informationen über die Situation des Lernenden versorgt werden; dies ergibt sich aus der konstruktivistischen Forderung nach Situierung der Lernprozesse (vgl. 4.3.1). Diese Zugriffsmöglichkeiten sollten dabei soweit wie möglich standardkonform eingerichtet werden.

AF7 Das System sollte für Lerngelegenheiten Navigationsmöglichkeiten anbieten, die zu anderen relevanten Lerngelegenheiten führen (z.B. zu Lernobjekten mögliche Ansprechpartner anbieten). Damit wird der Lernende in die Lage versetzt, die eigenen Lernprozesse selbst weiter zu steuern, so dass nicht der komplette Lernprozess vorstrukturiert wird, sondern wirklich nur ein »Gerüst« gebaut wird (vgl. Abschnitt 4.3.1).

- **Dokumentieren.**

AF6 Das System sollte die Möglichkeit bieten, die eigenen Lernaktivitäten zu dokumentieren.

7.1.2. Funktionale Anforderungen: Kontextaspekte

Zur Konkretisierung des Situationsbegriffs müssen noch relevante Aspekte identifiziert werden, die für die Lernunterstützung berücksichtigt werden müssen. Relevant ist ein Aspekt dann, wenn unter Kenntnis des Aspektes ein differenzierteres und besser angepasstes Systemverhalten möglich ist.

- AFK1 Wissensstand.** Ein sicherlich sehr wichtiger Aspekt ist der Wissensstand bzw. die vorhandenen Kompetenzen eines Mitarbeiters, weil sie die Grundlage für die Selektion liefern. Dies ergibt sich bereits aus dem Begriff der subjektiven Relevanz (vgl. Abschnitt 5.1.4), war aber auch wesentlicher Bestandteil der Informations(recherche)verhaltensmodelle z.B. von WILSON (Abschnitt 5.2.1 oder WANG BALDONADO (Abschnitt 5.3.2), da der Wissenstand bestimmt, welche Ressourcen verstanden werden können.
- AFK2 Aufgabe/Prozessschritt.** Die aktuelle Aufgabe bzw. der aktuelle Prozessschritt bildet die Grundlage für die Erfassung der Arbeitssituation, da der unmittelbare Wissensbedarf durch diese ausgelöst wird. Dieses Element wurde bereits im subjektiven Relevanzbegriff (vgl. Abschnitt 5.1.4) und von den geschäftsprozessorientierten Wissensmanagementansätzen (vgl. Abschnitt 4.3.2) betont.
- AFK3 Rolle/Organisationseinheit.** Neben der Ablauforganisation ist auch Einordnung in die Organisationsstruktur ein (statischerer) Bestandteil der Arbeitssituation, die mittels Einheiten wie Abteilungen oder Rollen beschrieben wird. Dies ergibt sich aus klassischen Anforderungsprofilen (wie sie auch im Kompetenzmanagement weiter verwendet werden).
- AFK4 Technische Ausstattung/Erreichbarkeit.** Weiterhin spielt es auch eine Rolle, ob die entsprechende Ressource sinnvoll genutzt werden kann. Hierfür ist es bei elektronischen Lernobjekten erforderlich, die technische Ausführungsumgebung (z.B. Lautsprecher, Bandbreite etc.) zu berücksichtigen. Für menschliche Lernressourcen ist beispielsweise auch deren momentane Erreichbarkeit einzubeziehen.
- AFK5 Lerncharakteristika.** Hierzu gehören die aus dem adaptiven E-Learning bekannten Lernstile und Lernertypen (vgl. z.B. [CMHE04]) sowie die facettierten Lernerpräferenzen (vgl. z.B. IMS Learning Information Profile, [IMS01]), die z.B. ausdrücken, welche Arten von Inhalten am besten geeignet sind. Hiermit gleichzusetzen (bei der Interpretation von Informationsrecherche als Lernprozess) sind die als »kognitive Stile« bezeichneten Elemente der Informationsverhaltensmodelle (z.B. von CHOO, vgl. Abschnitt 5.2.3).
- AFK6 Soziale Beziehungen.** Gerade für das informelle Lernen sind Informationen über die soziale Einbettung besonders wichtig. Denn als Kontaktpartner werden bevorzugt jene wahrgenommen, zu denen ein positiv belegtes Verhältnis besteht und keine Hierarchiebeziehung hemmend entgegensteht. Dies wurde beispielsweise in Zusammenarbeit mit dem Autor in [BS06b] und in [BS06a] in Bezugnahme auf Ergebnisse aus der HCI-Forschung ausführlich analysiert.

7.1.3. Funktionale Anforderungen: Lerngelegenheiten

Wie der Wissensreifungsprozess (vgl. Kapitel 3) gezeigt hat, greift ein auf pädagogisch aufbereitete Lernmaterialien beschränkter Ansatz insbesondere in betrieblichen Umgebungen zu kurz, da es ansonsten auf sehr allgemeine, dekontextualisierte Inhalte beschränkt ist. Dies entspricht auch nicht der Offenheit und Vielfalt der Lernaktivitäten, wie sie vom Konstruktivismus gefordert werden (vgl. Abschnitt 2.8.1); im Referenzmodell zu konstruktivistischen Lernumgebungen werden beispielsweise auch vergleichbare Fälle als Bestandteil der Lernumgebung gesehen. Dies führt dazu, dass im Gegensatz zu etlichen Forschungsansätzen zum rechnerunterstützten Lernen eine pädagogische Aufbereitung nicht voraussetzen können, sondern auch weniger »reife« Ressourcen (i.S.d. Wissensreifungsprozesses) zulassen und bewusst einbinden. Folgende Lerngelegenheiten müssen wir hierzu unterstützen:

AFL1 Lernobjekte. Dies sind didaktisch aufbereitete Ressourcen.

AFL2 Best Practices und andere »Gebrauchsdokumente«. Hierbei handelt es sich um Dokumente, die nicht primär zur Wissensvermittlung bzw. Kompetenzentwicklung gedacht sind, aber als Informationsartefakte dazu dienen können, Wissen und Kompetenz aufzubauen. Hierzu können auch vergleichbare Fälle, Checklisten o.ä. gehören.

AFL3 Kollegen. Um auch den direkten Austausch ohne vermittelnde Informationsartefakte zu unterstützen, müssen auch Kollegen als Lerngelegenheiten aufgefasst werden.

7.2. Nicht-funktionale Anforderungen

Neben den funktionalen Anforderungen beeinflussen auch die nicht-funktionalen Anforderungen wesentlich die Architektur der Lösung. Die nicht-funktionalen Eigenschaften ergeben sich dabei im wesentlichen aus den Ausführungen zum Wissensreifungsprozess als Makromodell für das arbeitsbegleitende Lernen sowie den in Kapitel 2 herausgestellten Eigenschaften:

AN1 Integrierbarkeit. Als Konsequenz aus der Einsicht, dass arbeitsbegleitende Lernprozesse nicht ablösbar aus der betrieblichen Situation zu betrachten sind (vgl. Abschnitt 2.7), kann es nicht das Ziel sein, für das situationsgesteuerte Lernen eine komplett neue Infrastruktur aufzubauen, sondern stattdessen gezielt existierende Infrastrukturen um spezielle Dienste zu ergänzen. Hierzu gehören insbesondere Geschäftsprozess-, Wissens- und Kompetenzmanagementsysteme.

AN2 Austauschbarkeit von Bausteinen. Auf technischer Ebene ergibt sich aus der Integrierbarkeit auch die Forderung, dass Implementierungen von Bausteinen des Sy-

stems gegen andere austauschbar sind, um sich in eine gegebene IT-Infrastruktur einzufügen.

AN3 Flexibilität für neue Lernunterstützungsformen. Kapitel 3 hat versucht, die Vielfalt der Lernformen aufzuspannen. Aus dieser Vielfalt wurde exemplarisch das Lernen bei Bedarf herausgegriffen, aber bereits in Abschnitt 2.5 wurde dargelegt, dass individuelle Lernprozesse zueinander in Beziehung stehen, so dass sie nicht sinnvoll isolierbar sind. Dementsprechend sollte die Infrastruktur vorbereitet sein, dass auch andere Lernformen unterstützt werden.

AN4 Hohe semantische Kohärenz. Die Analyse von Brüchen im Wissensreifungsprozess (Abschnitt 3.6) hat gezeigt, dass diese auf ein unterschiedliches Verständnis und unterschiedliche Konzeptualisierungen von Lernprozessen zurückzuführen sind. Um dies zu vermeiden, wird ein hohes Grad an semantischer Kohärenz des Systems gefordert. Komponenten eines Systems sind dabei **semantisch kohärent**, wenn sich das zugrundeliegende konzeptuelle Modell der Einzelkomponenten widerspruchsfrei in ein konzeptuelles Modell für das Gesamtsystem zusammenführen lässt. Ein semantischer Kohärenzgrad ergibt sich aus dem Vernetzungsgrad der einzelnen Teilmodelle miteinander.

7.3. Methodische Vorgehensweise

Wie sich schon aus der Komplexität des Phänomens Lernen und der in den nichtfunktionalen Anforderungen niedergeschlagenen Forderung nach starker Integration in die Arbeitsprozesse ergibt, kann die informationstechnische Lösung für das situationsgesteuerte Lernen nicht eine einzelne Anwendung sein, die alle Funktionen beinhaltet. Denn Integration bedeutet immer, dass die Lösung bis zu einem gewissen Grad spezifisch für das ist, in das hinein integriert wird. Sie wird für einen Mitarbeiter, dessen Arbeitsprozesse von der Interaktion mit spezifischen Fachanwendungen dominiert sind, anders aussehen, als für einen Wissensarbeiter, der sich einer Vielzahl von Instrumenten zur Kommunikation, Kollaboration, Artikulation und Suche bedient. Ebenso können wir nicht nur von einer Klasse von Anwendungen ausgehen: schon die Beschreibung der Systemfunktionalität zeigt, dass hier typische Benachrichtigungsanwendungen (z.T. integriert), Lernmanagementsysteme, Kommunikationswerkzeuge, Portfoliomanagement uvm. versammelt sind (die jeweils als eigener Anwendungstyp zu sehen sind), die lose miteinander gekoppelt erscheinen.

Diese Beobachtungen decken sich mit aktuellen Diskussionssträngen, die das bisherige »dominant design« von Lernumgebungen vor einer Ablösung durch einen Paradigmenwechsel sehen [WLB⁺06]. An die Stelle von integrierten Lernmanagementsystemen, die Lernen primär verwalten und organisieren, werden sog. »Personal Learning Environ-

ments« (PLE, [Att07]) treten, die Lernen unterstützen, indem sie es den Lernenden organisieren lassen. Ihr Systemparadigma sind dann auf technischer Ebene keine geschlossenen Anwendungen, sondern offene Sammlungen von Anwendungen, die *auch* (aber nicht nur) lernunterstützend wirken und die (im Gegensatz zum heutigen Desktop) über ein geteiltes Verständnis der Lerndomäne verfügen, so dass sie sich (lose) koppeln lassen.

Ausgehend von unserer bisherigen Analyse und der (noch vagen) Vision einer PLE ist das Ziel also eine domänenzentrierte Familie von Anwendungen für die (meist arbeitsbegleitende) Lernunterstützung, deren Konstruktion wir im Rahmen dieser Arbeit dadurch erleichtern wollen, dass wir einen wiederverwendbaren Kern bereitstellen (ein Dienstrahmenwerk). Wenn man für ein solches Systemverständnis ein Rahmenwerk konstruieren will, so handelt es sich hierbei nicht mehr um ein Anwendungsrahmenwerk, das sich auf die Bereitstellung von Funktionen für einen bestimmten Anwendungstypus beschränkt, sondern um ein Domänenrahmenwerk, das Familien von Anwendungen in einer bestimmten Domäne mit wiederverwendbaren Komponenten unterstützt (vgl. [GB01], [Pre97]). Im vorliegenden Fall ist die Domäne dabei die Lernunterstützung für situationsgesteuerte Lernprozesse am Arbeitsplatz und eine konkrete Anwendung ein lernunterstützendes System in einem bestimmten Unternehmen, eingebettet in dessen Organisationskontext.

Für die Aufstellung solcher Domänenrahmenwerke wurden unter dem Begriff »Domain Engineering«¹ Vorgehensmodelle entwickelt. Sie trennen die Entwicklung des Domänenrahmenwerks von der Entwicklung der Anwendungen und teilen den Entwurfsprozess in üblicherweise drei Schritte (vgl. z.B. [APD91], [BKPS04] und Abb. 7.1):

- Domänenanalyse (mit dem Ergebnis eines Anforderungskataloges)
 - Ermittlung der Anforderungen
 - Identifikation der Architekturtreiber als architekturelevante Anforderungen
 - Auswahl eines Architekturstils zur Umsetzung der Architekturtreiber
- Domänenentwurf (mit dem Ergebnis eines Domänenmodells und einer Referenzarchitektur) und
- Implementierung der Infrastruktur.

Partiell wurde die Domänenanalyse bereits in den vorangegangenen Kapiteln und am Anfang dieses Kapitels strukturiert gesammelt. Was jetzt noch fehlt, ist die Auswahl eines Architekturstils. Dies geschieht im folgenden Abschnitt in Form eines »ontologiezentrierten Architekturstils«. Er bildet die Basis für die Erarbeitung des Domänenmodells (Kapitel

¹Oft wird dieser Begriff übersetzt als »Software-Produktlinienentwicklung«, was aber m.E. die Problemstellung zu stark auf die Perspektive von Software-Unternehmen und ihren Produktstrategien reduziert (so z.B. [BKPS04]).

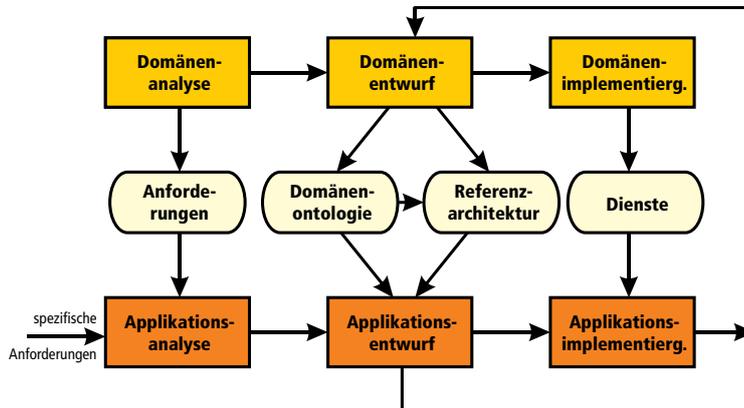


Abbildung 7.1.: Domänen- und Anwendungsentwicklungsprozesse, basierend auf [BKPS04]

8 und 9) als Domänenontologie. Dieser Architekturstil wird dann in Kapitel 10 zu einer Referenzarchitektur konkretisiert.



7.4. Ontologiezentrierter, dienstorientierter Architekturstil

Von den Anforderungen sind besonders die nicht-funktionalen Anforderungen architekturrelevant. Anforderungen AN2 und AN3 können nur dann erfüllt werden, wenn wir auf eine Architektur mit loser Kopplung statt eines monolithischen Systems setzen, so dass bedarfsorientiert Komponenten eingebunden werden können. In Verbindung mit Anforderung AN1 bietet sich hier das derzeit dominierende Paradigma der **dienstorientierten Architektur** an, das auf klare, umfassende Schnittstellendefinitionen setzt, um so lose Kopplung und flexible Rekonfiguration für unterschiedliche Anforderungen zu ermöglichen.

In komplexen Szenarien reichen allerdings Schnittstellen als Architekturgestaltungsmittel nicht mehr aus. Denn das Schnittstellenprinzip sagt noch nichts über das Zusammenwirken der Dienste aus. Auch wenn dies bislang in der Diskussion um dienstorientierte Architekturen vernachlässigt wird, kann nachhaltiges Zusammenwirken von Diensten (in Analogie zum Zusammenwirken von Menschen) nur gelingen, wenn sie über ein *geteiltes* Verständnis der zugrundeliegenden Domäne verfügen. Dies ist, was Anforderung AN4 mit dem Begriff der semantischen Kohärenz fordert und von EVANS als »shared kernel«-Muster bezeichnet wird; in Informationsintegrationsansätzen entspricht dies den »local as view«-Ansätzen [Ull97], die Quellen mit dem Vokabular des Domänenmodells beschreiben. Hierfür bieten sich Ontologien als »geteilte formale Konzeptualisierungen« [UG96] als Mittel der Wahl zur Festlegung der wesentlichen Begriffe und ihrer Beziehungen an. Im Sinne des Domain Engineering heißt das nichts anderes als ein ontologiebasiertes Domä-

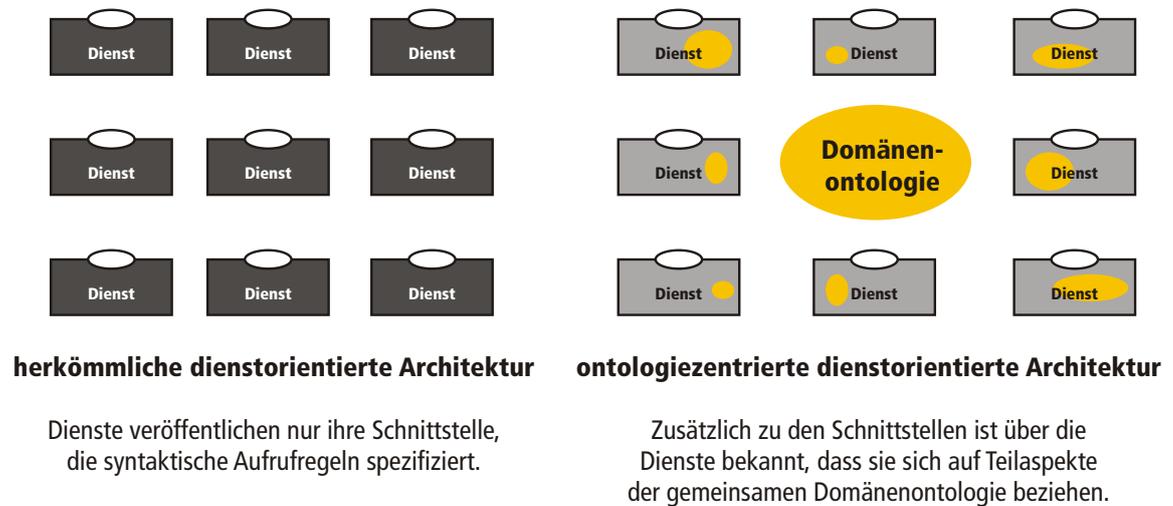


Abbildung 7.2.: Klassischer dienstorientierter Architekturstil vs. ontologiezentrierter dienstorientierter Architekturstil

nenmodell oder kurz eine »Domänenontologie« (als »grundlegendes Vokabular für einen bestimmten Anwendungsbereich« [Gua95]).

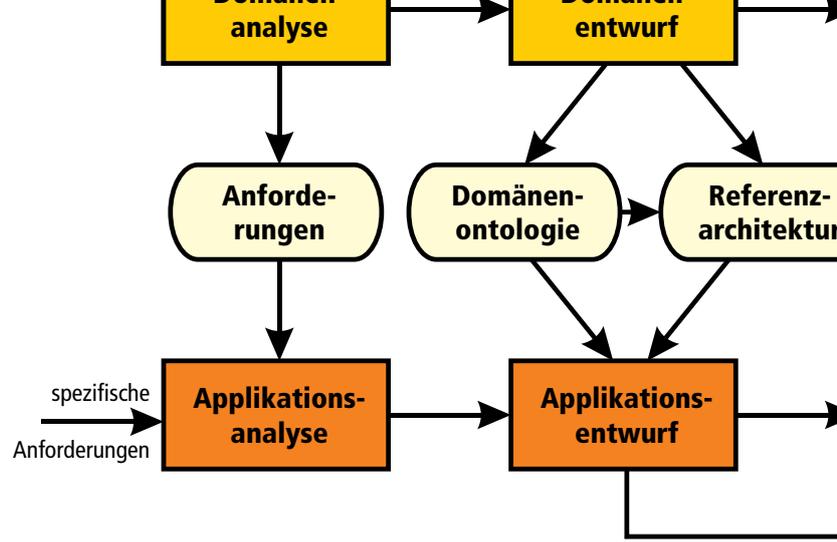
Ontologien können durch ihre Formalisierung nicht nur in der Entwurfs- und Modellierungsphase genutzt werden, sondern auch als Bezugspunkt in der operativen Implementierung (vgl. [AFGD02]). Hierdurch lässt sich ein »Ontology-Based Domain-Driven Design« realisieren [Hru05]: Die bei der Domänenanalyse entstehenden Modelle werden maschinenverarbeitbar kodiert (was durch Ontologien mit formaler Semantik gelingt); die Implementierung wird dann entweder (in Sinne eines modellgetriebenen Ansatzes) aus diesem Modell zur Übersetzungszeit partiell generiert oder die Implementierung greift auf die Ontologie zur Laufzeit zu [Hru05]. Dadurch wird (a) der Nutzen des Domänenmodells erhöht, (b) die semantische Kohärenz der einzelnen Bausteine des Rahmenwerks erhöht und (c) bei der Weiterentwicklung die Konsistenz sichergestellt.

Der wesentliche Vorteil dieses ontologiezentrierten Ansatzes ist hierbei (vgl. Abb. 7.2), dass das Instrument der Schnittstellendefinitionen trotz aller Bemühungen in Richtung »Semantic Web Services« wie in Programmiersprachen immer noch primär auf der syntaktischen Ebene verbleibt, so dass sich die geforderte semantische Kohärenz dadurch auch nicht ansatzweise garantieren lässt, während mit der gemeinsamen Domänenontologie, von der Teile jeweils möglichst tief in der operationalen Dienstleistung verankert werden sollte, ein Instrument existiert, das eine gemeinsame Semantik zu repräsentieren und zu nutzen hilft. Dies bedeutet sicherlich einen Eingriff in die Autonomie der Dienste; umgekehrt reduziert sich die Komplexität der Integration erheblich.

7.5. Fazit

Mit dem ontologiezentrierten Architekturstil auf der Basis eines Domain-Engineering-Ansatzes ist das weitere Vorgehen vorgezeichnet. Auf der Basis der aus den vorangegangenen Teilen abgeleiteten Anforderungen geht es nun zunächst darum, eine Domänenontologie als übergreifende Konzeptualisierung zu entwickeln, bevor im nächsten Schritt anhand dieser Domänenontologie Dienste identifiziert und in einer Referenzarchitektur angeordnet werden.

8.



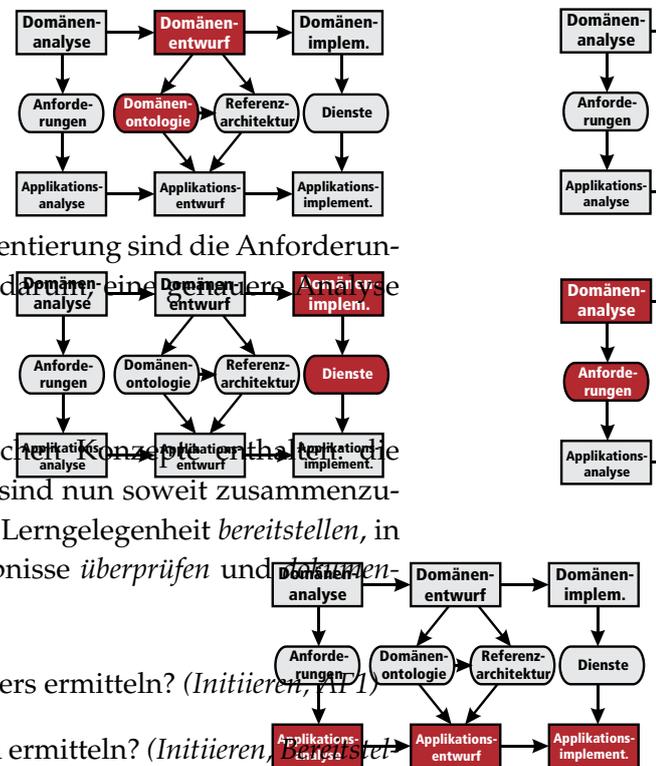
Domänenontologie

Im vorangegangenen Kapitel wurde die Entwicklungsmethodik grob skizziert. Der ontologiezentrierte Architekturstil weist dabei neben dem zu unterstützten Prozessmodell des situationgesteuerten Lernen der Domänenontologie eine dominierende Rolle zu. In diesem Kapitel soll es nun darum gehen, wie eine solche Ontologie auszusehen hat. Ausgangspunkt hierfür und Orientierung sind die Anforderungen aus dem vorangegangenen Kapitel. Hier nun geht es darum, eine Domänenontologie durchzuführen und ein geeignetes Modell zu finden.

8.1. Grundlegende Überlegungen

In den Anforderungen sind bereits die beiden wesentlichen Konzepte des situationgesteuerten Lernen, die Mitarbeiter/Lernenden und die Lerngelegenheiten. Beide sind nun soweit zusammenzubringen, dass das System Lernprozesse *initiiieren*, gewählte Lerngelegenheit *bereitstellen*, in Lernprozessen *unterstützen*, die entsprechenden Lernergebnisse *überprüfen* und *dokumentieren* kann. Hierbei geht es um folgende Kernfragen:

- Wie lässt sich der aktuelle Lernbedarf eines Mitarbeiters ermitteln? (*Initiiieren*, AF1)
- Wie lassen sich subjektiv relevante Lerngelegenheiten ermitteln? (*Initiiieren*, *Bereitstellen*, AF2, AF4)
- Wie lässt sich Art und Zeitpunkt der Empfehlung so gestalten, dass diese für den potentiell Lernenden nicht als störend wahrgenommen wird? (*Bereitstellen*, AF3, AF5)
- Wie lassen sich während des Lernprozesses weitere Schritte des Lernprozesses gestalten? (*Unterstützen*, AF6, AF7)



8.1.1. Kompetenzorientierter Grundansatz

Eine wesentliche Weichenstellung bei der Ontologieentwicklung ist die Frage, welches Paradigma zur Ermittlung eines Lernbedarfs gewählt werden soll. Die existierenden Ansätze hierzu lassen sich in vier Klassen einteilen:

- Syntaktische Ansätze
- Geschäftsprozessorientierte Ansätze
- Wissensorientierte Ansätze
- Kompetenzorientierte Ansätze

Diese Ansätze sollen mit ihren Hauptvertretern kurz vorgestellt werden.

Syntaktische Ansätze

Im *Just-in-Time Information Retrieval* (vgl. Abschnitt 5.5) wird versucht, das Problem auf syntaktischer Ebene anzugehen, indem als wichtige eingeschätzte Stichwörter im aktuellen Aufgabenkontext (z.B. in einer Textverarbeitung, einem Web-Browser o.ä., vgl. [Rho00]) als Basis für eine Anfrage an ein Information-Retrieval-System genutzt werden. Neuere Ansätze (die sich interessanterweise auch als Lernunterstützung verstehen) versuchen, Stichwörter aus Aufgabenbeschreibungen auf persönlichen Aufgabenlisten für diesen Zweck zu nutzen [RS06].

Diese Ansätze können zwar ohne eine zusätzliche Modellierung auskommen; jedoch treten die typischen Probleme eines auf die syntaktische Ebene beschränkten Ansatzes auf: mehrdeutige oder synonyme Begriffe sowie die starke Abhängigkeit von einer textuellen Umgebung (textuelle Beschreibungen zur Extraktion von Stichwörtern müssen existieren). Durch Techniken wie Text Mining lässt sich zwar bis zu einem gewissen Grad sowohl die Aufgabenbeschreibung als auch die Lernobjektbeschreibung auf eine »semantische« Ebene heben; diese Verfahren haben allerdings bislang – bis auf Spezialumgebungen – keine zufriedenstellende Zuverlässigkeit.

Geschäftsprozessorientierte Ansätze

In den Ansätzen des *geschäftsprozessorientierten Wissensmanagements* (vgl. Abschnitt 4.3.2) wird eine symbolische Repräsentationsschicht dazwischen geschoben: Repräsentationen von Geschäftsprozessen und ihren Aktivitäten. Hierbei ist die Grundannahme, dass sich Informationsartefakte direkt Prozessaktivitäten oder Aufgaben zuordnen lassen. Diese direkte Zuordnung ist allerdings problematisch:

- Es wird vorausgesetzt, dass Prozesse so strukturiert und modelliert sind, dass sie unterschiedliches fachliches Wissen erfordern. In vielen Fällen ist diese Annahme nicht angebracht (vgl. hierzu z.B. die Ausführungen in [BHS07]), vielmehr sollten die Orientierung an methodischen Kriterien erfolgen (z.B. Problemlösungsprozesse).
- Bei sich häufig ändernden Prozessen und Prozessaktivitäten ist unklar, welche Zuordnungen übernommen werden können, weil sie noch immer relevante Beziehungen ausdrücken, und welche hinfällig sind. Ein Beispiel hierbei ist ein Überblicksdokument, das den Prozess erklärt, das für einen geänderten Prozess aber nicht mehr geeignet ist. Demgegenüber kann prozessunabhängiges Wissen über dahinter stehende Zusammenhänge weiterhin verwendet werden.

Wissensorientierte Ansätze

Geht man stattdessen von bereits teilweise formalisierten Bedürfnissen (in Form von Textfragmenten oder Beschreibungen) vom Individuum aus, so ist man bei wissensorientierten Ansätzen. Besonders im Bereich der Intelligenten Tutoriellen Systeme (vgl. als Überblick [BGMU06]), aber auch in anderen Ansätzen (z.B. zur Kursgenerierung, vgl. [AL06]) wird auf Wissensstrukturen gesetzt, die das zu erlernende Wissensgebiet mit seinen inneren Abhängigkeiten modellieren. Diese Ansätze haben den Vorteil, dass auf beiden Seiten (Lerner und Lerngelegenheiten) auf semantisch eindeutige Repräsentationen gesetzt wird, wenn man dabei auch in Kauf nimmt, dass eine zusätzliche Modellierung erforderlich ist.

Kompetenzorientierte Ansätze

Kompetenzorientierte Ansätze (wie vom Autor in [KS07] beschrieben, ähnlich [Woe02]) sind mit den wissensorientierten Ansätzen eng verwandt. Im Unterschied zu den wissensorientierten Ansätzen versuchen sie, sich stärker an der klassischen Personalentwicklung und ihrem Fokus auf das Individuum aus organisationaler Sicht zu orientieren. Hier haben sich Kompetenzen als »operationalisierbare Konstruktionen menschlicher Leistung« [SSR04] (ähnlich [Kir04]) etabliert. Sie dienen zur Beschreibung dessen, was ein Mensch »kann«. Dies umfasst mehr als das bloße Wissen, sondern beschreibt auch, wie der einzelne in der Lage ist, das Wissen in die Anwendung zu bringen. Darüber hinaus sind in Kompetenzen auch Fähigkeiten und Fertigkeiten enthalten (vgl. [Kun05]).

Grundidee für die Ermittlung des Lernbedarfs ist hierbei, dass Kompetenzen als Brücke zwischen dem einzelnen Mitarbeiter und den organisationalen Anforderungen (die den Lernbedarf bestimmen) dienen: der Mitarbeiter wird beschrieben durch die Kompetenzen, über die er verfügt; Organisationsstrukturen werden durch Kompetenzanforderungen an-notiert.

Fazit: Kompetenzorientierter Grundansatz

Für diese Arbeit erscheint der kompetenzorientierte Grundansatz der vielversprechendste:

- Der Kompetenzbegriff ist ganzheitlicher angelegt als der Wissensbegriff, so dass dadurch eine zu einseitige Sicht auf Lernen vermieden wird.
- Die Kompetenzorientierung bietet die Möglichkeit, den Ansatz mit anderen längerfristigen Personalentwicklungsaktivitäten zu integrieren (die insbesondere auch persönliche Entwicklungsziele mit einbeziehen), wie in Kapitel 2 ausgeführt.
- Im Gegensatz zum geschäftsprozessorientierten Ansatz verspricht die Indirektion (kein direktes Verbinden von Lerngelegenheiten mit Organisationsstrukturen, sondern indirekt über Kompetenzen) eine stabilere Modellierung, da Kompetenzmodelle grundsätzlich als stabiler angenommen werden können als Aufbau- und Ablauforganisation (vgl. [Bie06]). Weiterhin zeigt auch die Vielfalt der Kontextmerkmale, die den Lernbedarf beeinflussen können, dass es nicht sinnvoll ist, sich auf bestimmte Entitäten wie Prozesse festzulegen, sondern diese eher als Hilfsmittel zur Ermittlung eines Kompetenzbedarfs heranzuziehen.
- Im Gegensatz zu syntaktischen Ansätzen ist es darüber hinaus auch möglich, sowohl Informationsartefakte (insbesondere Lernobjekte) als auch Kollegen auf einheitliche Weise als Lerngelegenheiten zu betrachten (Lernobjekte wurden dazu erstellt, Kompetenzen zu vermitteln, und Menschen verfügen über Kompetenzen, die sie weitergeben können).

Mit dieser Weichenstellung kann nun damit begonnen werden, zunächst beim Lernenden als zentralem Akteur und seiner Situation mit einer weiteren Formalisierung zu beginnen, bevor im darauffolgenden Abschnitt Lerngelegenheiten näher charakterisiert werden.

8.2. Formalisierung des Kontextbegriffes und Kontextontologie

Bislang wurde Kontext informell als technische Operationalisierung des Situationsbegriffes eingeführt. In diesem Abschnitt geht es nun darum, diesen im Rahmen der Ontologieentwicklung zu präzisieren, zu formalisieren und durch Auffächerung in seine Bestandteile zu operationalisieren.

8.2.1. Grundlegendes zum Kontextbegriff

»Kontext« wird in unterschiedlichen Bereichen mit wechselnder Bedeutung verwendet (vgl. für einen exemplarischen Überblick [Hen03] oder [Bré99]). Als semantischen Kern

haben BAZIRE & UND BRÉZILLON [BB05] in einer systematischen Analyse von insgesamt rund 150 Definitionen herausgearbeitet: Kontext ist eine Menge von Einschränkungen, die das Verhalten eines Systems (was sowohl einen Benutzer oder ein Rechnersystem bezeichnen kann) in einer bestimmten Aufgabe beeinflussen kann. In dieser Arbeit basiert die Verwendung des Begriffes »Kontext« auf Arbeiten im Umfeld des »ubiquitous« bzw. »pervasive computing«. Denn auch dort steht ein situationsbewusstes Systemverhalten im Vordergrund, wobei dies in dieser Arbeit natürlicherweise auf die Situation eines menschlichen Benutzers beschränkt ist. Grundsätzlich gibt es hier zwei Ansätze zur Definition dessen, was Kontext ist: intensionale und extensionale Ansätze. Beim **extensionalen Ansatz** wird der Begriff durch Aufzählung der einzelnen Bestandteile definiert wie z.B. Ort, umgebende Personen und Objekte (so z.B. [Sch95], [RPM97], [Dey98], [Sch02b], [Mit02]). Beim **intensionalen Ansatz** wird versucht, eine von den einzelnen Bestandteilen abstrahierende, allgemeine Definition zu geben, die dann auch unabhängig vom jeweiligen Anwendungsgebiet ist. Hier hat sich inzwischen eine allgemein akzeptierte Definition des Begriffes etabliert [Dey01]:

*Context is any information that can be used to characterize the situation of an entity.
An entity is a person, place, or object that is considered relevant to the interaction
between a user and an application, including the user and application themselves.*

Ähnlich sind u.a. auch die Definitionen von [CK00], [Cha02] oder [PNS⁺00]. In diesem Sinne soll für diese Arbeit auch folgende, auf dem Situationsbegriff aufbauende Definition gelten ([Bro96]):

Definition 8.1 (Kontext) *Ein Kontext ist das systemseitige Abbild einer Situation; ein Benutzungskontext (auch Nutzungskontext oder Benutzerkontext) ist das systemseitige Abbildung einer Benutzungssituation.*

Um besser über die »Bestandteile« eines Benutzungskontextes reden zu können, wird der folgende Begriff des Kontextmerkmals eingeführt :

Definition 8.2 (Kontextmerkmal) *Ein Kontextmerkmal ist eine konzeptuelle Eigenschaft zur näheren Charakterisierung des Benutzungskontextes. Die Ausprägungen dieser Eigenschaft heißen Kontextmerkmalsausprägungen (vgl. [Cha02]).*

Im Gegensatz zum informellen und unscharfen Begriff des Aspekts (wie er im vorangegangenen Kapitel verwendet wurde) sind Kontextmerkmale wohldefinierte Konstrukte zur näheren Beschreibung eines oder ggf. auch mehrerer Aspekte; Merkmale lassen sich also als ein Modell für Situationsaspekte verstehen.¹ Dies entspricht in einer objektorientierten Modellierung Attributen oder Assoziationen, die vom Benutzer ausgehen.

¹Bei STRANG [Str03] findet man den Begriff des »Aspekts« von Kontextinformationen und bei KLEMKE [Kle02] den Begriff der »Kontextdimension« anstelle von »Kontextmerkmal«.

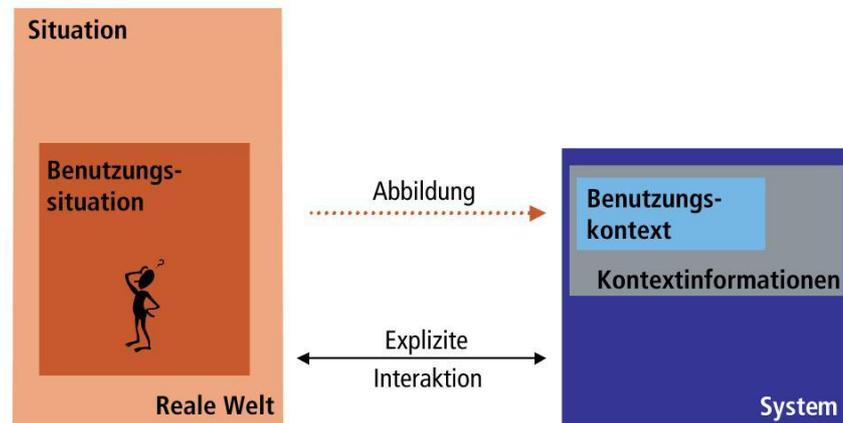


Abbildung 8.1.: Benutzungskontext und Benutzersituation

Während also die »Benutzungssituation« den Ausschnitt der realen Welt repräsentiert, der für die Interaktion eines Benutzers mit einem System relevant ist und sich mittels Situationsaspekten beschreiben lässt, ist der »Benutzungskontext« das systemseitige Abbild der Benutzungssituation (vgl. Abb. 8.1), das mit Hilfe von Kontextmerkmalen beschrieben wird.²

Diese Abgrenzung lässt sich auch auf Dienste übertragen. Dienste heißen **situationsbewusst**, falls sie durch den Einsatz von kontextbewussten Verfahren die Interaktion eines Systems mit dem Benutzer effizienter gestalten. Dabei heißt ein Verfahren **kontextbewusst**, falls es über einen Zugang zum jeweils aktuellen Benutzungskontext verfügt und durch diesen beeinflusst wird. Hierbei ist bedeutsam, dass »situationsbewusst« durchaus fordert, dass die Verwendung eines Benutzungskontextes zu positiven Ergebnissen für den Benutzer führt, während »kontextbewusst« nur besagt, dass das Verfahren den Kontext kennt und ihn nicht ignoriert.

8.2.2. Existierende Ansätze zur Identifikation und Klassifikation von Kontextmerkmalen

Nachdem die grundlegenden Begriffe »Kontext« und »Kontextmerkmal« eingeführt sind, geht es im nächsten Schritt darum, Kontextmerkmale zur näheren Charakterisierung zu identifizieren und sinnvoll zu ordnen. Hierfür dient neben den Anforderungen aus Abschnitt 7.1 eine Betrachtung existierender Modellierungsansätze, die unter unterschiedli-

²Eine ähnliche Abgrenzung zwischen Kontext und Situation findet sich auch in in [SG01]. KLEMKER ([Kle02]) differenziert ähnlich, benutzt allerdings »Kontext« für Situation und »Kontextmodell« für Kontext.

chen Begriffen wie »Kontextmodell«, »Kontextkategorien«, »Klassifikation« oder »Taxonomien von Kontextmerkmalen« o.ä. zu finden sind und in den folgenden Unterabschnitten kurz betrachtet werden.

Erste phänomenologische Ansätze aus dem Bereich des »Ubiquitous Computing«

Die ersten Begriffsbildungen definierten Kontext rein extensional über die betrachteten Merkmale; entsprechend gibt es zahlreiche Ansätze, Kontext zu klassifizieren, wobei ein starker Telematik-Hintergrund zu beobachten ist. SCHILIT [SAW94] differenziert zwischen technischem Kontext (Netzwerkzugang, Netzwerkbandbreite, Verbindungskosten, verfügbare Ressourcen wie Prozessoren, Drucker und Bildschirme), Benutzerkontext (Aufenthaltort, Personen in der näheren Umgebung) sowie physikalischem Kontext (Lichtverhältnisse, Geräuschpegel und Temperatur). Eine Weiterentwicklung ist der Ansatz von SCHMIDT ET AL. [SBG98], der mit den beiden Grundkategorien menschliche und physikalische Faktoren arbeitet; menschliche Faktoren werden dabei weiter unterteilt in Benutzer, soziale Umgebung und Aufgabe, während die physikalischen Faktoren unterteilt werden in Umgebungsbedingungen, Infrastruktur und Ort. DEY [Dey01] führt die beiden Kategorien primärer und sekundärer Kontext ein, wobei der primäre Kontext aus Identität, Tätigkeit, Zeit und Aufenthaltsort besteht, während sich der sekundäre Kontext aus dem primären Kontext und evtl. Hintergrundwissen ableiten läßt.

Ansätze aus dem Bereich der Benutzermodellierung

An den frühen Ansätzen aus dem *Ubiquitous Computing* lässt sich ablesen, dass die Vorstellung dominiert, dass »Kontext« das ist, was den Benutzer umgibt, während Eigenschaften des Benutzers wie Vorwissen, Interessen, Ziele, Stresszustand etc. meist unberücksichtigt bleiben (so auch [Jam01]). Letzteres wurde in der Vergangenheit unter dem Begriff »Benutzerprofil« oder »Benutzermodell« [Jam99] betrachtet, die zur Realisierung von adaptiven Systemen dienen und bereits eine lange Tradition haben. Ein **Benutzermodell** ist dabei ein detailliertes, explizites, maschinenverarbeitbares Modell des einzelnen Nutzers, um auf dessen Basis das Systemverhalten anpassen zu können. Im Gegensatz zum Kontextbegriff sind Struktur und Semantik solcher Modelle traditionell auf die einzelne Anwendung beschränkt, wenn auch neuere Ansätze wie z.B. [HK03] die Austauschbarkeit solcher Benutzermodelle zwischen Anwendungen angehen. Insoweit sollte man den Kontextbegriff als Erweiterung und Verallgemeinerung ansehen, wobei zu beachten ist, daß Kontext eine stärker zeitabhängige Komponente hat und durchaus auch flüchtigere Bestandteile haben kann. Allerdings ist das Verhältnis zwischen Benutzermodell und Benutzungssituation/-kontext in der Literatur nicht eindeutig. So verwendet beispielsweise JAMESON [Jam01] den Begriff Kontext im wesentlichen für die Umgebung und stellt das Benutzermodell

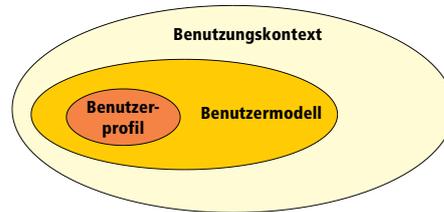


Abbildung 8.2.: Benutzungssituation, Benutzermodell und Benutzerprofil

dem gegenüber. Allerdings ist der Schlussfolgerung beizupflichten, dass beides zusammengehört. GÖKER UND MYRHAUG[GM02] sehen im Situations-/Kontextbegriff das allgemeinere Konzept, das neben unmittelbar benutzerbezogenen Merkmalen auch weitergehende Merkmale in der Umgebung zu fassen vermag. Das **Benutzerprofil** bezeichnet einen sehr eingeschränkten Teil des Benutzungskontextes. Hier werden bestimmte Informationen über den Benutzer explizit und persistent gespeichert. Hierbei geht man meist von einer gewissen Ausmittlung aus. Die einzelnen Bestandteile werden als erheblich stabiler angenommen (z.B. Interessen); anders als beim Kontextbegriff spielt die Zeit eine viel geringere Rolle. Dies schließt klar eine dynamische Ermittlung, z.B. mittels Sensoren, aus. Insgesamt wird ein Benutzerprofil oft als ein Element des Benutzermodells gesehen [GK03].

Damit lässt sich die Sichtweise der Arbeit in Abb. 8.2 zusammenfassen. Daraus ergibt sich auch, dass der Forschungszweig der Benutzermodellierung wichtige Beiträge zu liefern vermag.

Eine Bestandsaufnahme von dem, was über Spezialanwendungen hinaus in Benutzermodellen üblicherweise enthalten ist, liefern KOBZA ET AL. [KKP01]. Sie differenzieren zunächst einmal zwischen »Benutzerdaten« (*user data*) und »Benutzungsdaten« (*usage data*). Während ersteres alle persönlichen Eigenschaften des Benutzers beschreibt, bezieht sich das zweite auf das (Interaktions-)Verhalten des Benutzers.

Zu den *Benutzerdaten* zählen sie:

- Demographische Daten (Kontaktdaten, geographische Daten, Eigenschaften wie Einkommen, Geschlecht etc., psychographische Daten, Kundendaten, Registrierungen)
- Wissen des Benutzers
- Fähigkeiten und Fertigkeiten des Benutzers
- Benutzerinteressen und -präferenzen
- Ziele und Pläne

Als *Benutzungsdaten* haben sie identifiziert:

- beobachtbare Nutzung
 - ausgewählte Aktionen
 - Zeitliches Betrachtungsverhalten
 - Bewertungen
 - Käufe und kaufbezogenes Verhalten
 - weitere bestätigende oder widerlegende Aktionen
- Regelmäßigkeiten in der Benutzung
 - Häufigkeit
 - Korrelationen zwischen Situation und Aktion
 - Handlungssequenzen

Weiterhin betrachten sie auch die *Umgebung* (also den außerhalb des Benutzermodells liegenden Teil des Kontexts):

- Softwareumgebung (Browserversion und Plattform, verfügbare Plugins, Java und Javascript)
- Hardwareumgebung (Bandbreite, Prozessorgeschwindigkeit, Anzeigegeräte, Eingabegeräte)
- Ort (Aufenthaltsort und dessen Eigenschaften)

Aus Sicht der Benutzermodellierung macht die Abgrenzung zwischen Benutzerdaten und Benutzungsdaten durchaus Sinn, doch je mehr man in komplexe Umgebungen gerät, desto mehr verschwindet die Grenze zwischen beiden, so dass wir diese Differenzierung im weiteren Verlauf der Arbeit nicht weiter verfolgen. So ist beispielsweise die Grenze zwischen dem beobachteten Verhalten und den Zielen und Plänen fließend; hier dienen beispielsweise die erwähnten Korrelationen als Heuristiken. Die identifizierten Eigenschaften sind allerdings eine wichtige Grundlage für die Kontextontologie.

Benutzermodellierung im E-Learning

Aus dem Bereich der domänenspezifischen Benutzermodellierungsansätze ist besonders der Bereich des adaptiven E-Learning interessant. Adaptivität als Anpassung des Systemverhaltens an den jeweiligen Benutzer spielt in rechnerunterstützten Lernumgebungen schon immer eine große Rolle (vgl. z.B. [Bru98], [HH98], [See03]), was nicht zuletzt darin

begründet ist, dass »Führung« im pädagogischen Sinne dann besonders effektiv ist, wenn sie auf den jeweiligen Lernenden eingeht. Die Formen von Adaptivität sind dabei höchst unterschiedlich und lassen sich grob in makroadaptive und mikroadaptive Ansätze aufteilen [PL04]. Während makroadaptive Verfahren auf der Basis einer groben Klassifikation der Lernenden (Altersgruppe, Bildungsniveau) und Vorwissen/Leistungen auf der Granularitätsebene von Kursen arbeiten und somit im wesentlichen die Selektion von passendem Lernmaterial (auf der Basis von Lernzielen, Detaillierungsgrad etc.) betreffen, konzentrieren sich mikroadaptive Verfahren auf die durchgehende Begleitung des Lernprozesses in seinen Einzelschritten. Hierbei müssen mikroadaptive Verfahren auf weitaus feingranularere Informationen über den Lernenden zurückgreifen. Typisch hierfür sind Leistungen oder Fehler bei Übungen/Tests, wie es von sog. »Intelligenten tutoriellen Systemen« praktiziert wird. Zwischen makro- und mikroadaptiven Verfahren liegen die sog. »Aptitude-treatment Interaction« (ATI), bei der auf der Basis von Lernercharakteristika die optimale Lehrstrategie ausgewählt wird, wobei diese Auswahl oft auf der Basis empirischer Ergebnisse bzgl. der Lerneffizienz bestimmter Typen von Lernenden erfolgt.

Mikroadaptive Ansätze sind für unsere Kontextontologie weniger interessant, da sie i.d.R. sehr spezifische Kontextinformationen erfordern, die nur innerhalb des jeweiligen Lernobjektes relevant sind (z.B. die Bildschirmposition, auf die ein Benutzer seine Aufmerksamkeit richtet [GPT⁺05]). Makroadaptive und ATI-Verfahren sind bereits in entsprechende Standards wie PAPI [IEE02] oder Learning Information Profile (LIP) [IMS01] eingeflossen [DN03]. Dementsprechend können diese beiden stellvertretend hier analysiert werden. In seinem Kern spezifiziert PAPI sechs Blöcke:

- *Lernerpräferenzen.* Hierunter sind Präferenzen des Lernenden bezeichnet, die die Mensch-Maschine-Interaktion verbessern, also z.B. bevorzugte Ein-/Ausgabegeräte.
- *Leistungsinformationen.* Hier werden primär durch das Lernmanagement alle Informationen zu vergangenen und aktuellen Aktivitäten sowie Zielen für die Zukunft gespeichert.
- *Portfolioinformationen.* Dieser Block greift die oben schon erwähnte Idee der elektronischen Portfolios auf und enthält Referenzen auf Arbeiten und Ergebnisse des Lernenden, z.B. bestandene Prüfungen, Aufsätze etc.
- *Beziehungsinformationen.* In diesem Block werden die Beziehungen zu anderen Personen wie z.B. Lehrern, Tutoren und anderen Lernenden gespeichert.
- *Sicherheitsinformationen.* Hier können alle Authentisierungsinformationen wie z.B. Passwörter, öffentliche und private Schlüssel abgelegt werden.
- *Kontaktinformationen.* Dieser Block enthält alle Kontaktdaten (Adresse, Telefon etc.).

Damit beschränkt sich der Ansatz auf typische Aufgaben eines Lernmanagementsystems, was jedoch für eine Einbettung in die Arbeitsprozesse nicht ausreicht. IMS LIP ([IMS01], vgl. auch [DHNS04]) ist ein vergleichbarer Ansatz im Kontext von SCORM³, der sich auch weitgehend informationserhaltend auf IEEE PAPI abbilden läßt.

Persönliches Informationsmanagement

Ein anderer Bereich, der für die Arbeit relevant ist, ist der Bereich des persönlichen Informationsmanagements, der die Unterstützung der alltäglichen Arbeit (besonders für sog. »Wissensarbeiter«) anstrebt. Dementsprechend wird hier der Frage nachgegangen, wie man die Arbeitssituation beschreiben kann. Ein aktueller Ansatz ist hier das EPOS-Projekt [Sch05b], das als Kontextkategorien die folgenden identifiziert: Operational (geöffnete Anwendungen), Organisational, Umgebung, Historie, Aufmerksamkeit, Verhalten, Kausal (Aufgaben), Information.

Wissensmanagement

Für das Wissensmanagement im Unternehmensumfeld identifizieren GROSS UND KLEME [GK03] die Kategorien Organisational (Ablauf-, Aufbauorganisation), Domäne/Inhalt, Persönlich und Physisch. ENGELBACH & DELP [ED03] identifizieren auf der Basis einer empirischen Untersuchung in Unternehmen die relevanten Kontextfaktoren, die einen Einfluss auf die Informationsrecherche in Unternehmen haben. Das Ergebnis sind die vier Kategorien Person (Qualifikation, Erfahrung, Werte, Einstellungen, Lernmethodik), Rolle (Befugnisse und Rechte, Aufgaben und Pflichten, aufbauorganisatorische Einbindung, Verantwortung), Aufgabe (Prozesseinbettung, Formalisierungsgrad, Prozesshäufigkeit, Zeitbudget) und die Infrastruktur (Ort, Technik).

Das Konzept der *Wissenslage* von MAIER UND HÄDRICH ([Mai05], [HP05]) beschäftigt sich mit Situationen, in der von arbeitsorientierten Prozessen in wissensorientierte Prozesse übergegangen wird. Hierbei ist eine Wissenslage eine Klasse von Situationen, die durch Anlass (occasion), sog. »Kontext« und Modus gekennzeichnet sind, die einen solchen Übergang auslösen. Der *Anlass* bezeichnet die Art der Gelegenheit, an den Unternehmenszielen orientiert Wissen zu generieren oder zu erwerben. Der *Kontext* operationalisiert die Situation entlang von Organisationseinheiten, Rollen, Gruppen, aber auch die Kompetenzanforderungen und sozialen Beziehungen, wobei keine genaueren Angaben

³SCORM (*Sharable Content Object Reference Model*) ist ein De-Facto-Standard für interoperable (web-basierte) Lernressourcen (WBTs). Die Spezifikation, die federführend durch die *Advanced Distributed Learning Initiative* (ADL) entwickelt wird, spezifiziert im wesentlichen ein Paketformat (einschließlich Metadaten für Lernobjekte), eine Ablaufspezifikation sowie eine Laufzeitumgebung, die von Lernmanagementsystemen zu implementieren ist.

zum verwendeten Kontextmodell gemacht werden. Der *Modus* beschreibt die Arten von Aktivitäten.

ABECKER identifiziert im Geschäftsprozessorientierten Wissensmanagement drei Kontextkategorien für ein Informationsbedürfnis: Aufgabe, Rolle und Person [Abe04].

Integrative Ansätze

Neuere Ansätze zur Kontextmodellierung bzw. Benutzermodellierung (dort meist als »Ubiquitous User Modeling« bezeichnet [Hec06b]) überwinden meist die Trennung zwischen Benutzermodell und Kontext und teilen damit die Sichtweise der Arbeit, dass Kontext eine Verallgemeinerung des Benutzermodells ist. Insbesondere haben diese Ansätze auch erkannt, dass sich Kontextmodellierung nicht auf generischer Ebene lösen lässt, sondern viel stärker domänen- und problemspezifisch angegangen werden muss. In der *General User Modeling Ontology* (GUMO, [HSB⁺05]) (die mehr als nur Kontextkategorien und -merkmale enthält) wurden hierfür die generischen Kategorien Physisch, Räumlich, Zeitlich, Aktivität, Situation und Inferenz eingeführt, wobei die letzten beiden Konzepte zur weitergehenden Nutzung von Kontextinformation bereitstellen.

Ein Ansatz zu systematischen Ableitung solcher Kategorien stellt der Ansatz von KAEN-AMPORN PAN ([KO04], [KO05]) dar, bei dem die Kategorien aus der Aktivitätstheorie abgeleitet werden, die die menschliche Handlung durch die folgenden Elemente beschreibt: Subjekt, Objekt, Ergebnis, Werkzeuge, Gemeinschaft, Arbeitsteilung und Regeln.

8.2.3. Entwicklung der Kontextontologie

Im folgenden wird die Kontextontologie in zwei Schritten entwickelt: im ersten Schritt werden die Top-Level-Strukturen (Kontextkategorien) identifiziert, im zweiten Schritt geht es um die Konkretisierung in Form von Kontextmerkmalen.

Top-Level-Strukturen der Kontextontologie

Während der Ansatz der *General User Modeling Ontology* [HSB⁺05] Kontext über die in ihm vorkommenden Arten von Objekten bzw. abstrakten Konzepten zu klassifizieren versucht (z.B. Benutzer, Gerät, Ort, Zeit, Ereignis, Situation) und somit eine eher globale Sichtweise einnimmt, beschränkt sich die Kontextontologie auf eine benutzerzentrierte Sicht, d.h. Kontext wird durch zeitabhängige Eigenschaften des Benutzers charakterisiert. Wo andere Entitäten und deren Eigenschaften relevant sind (also z.B. Prozesse, Aufgaben u.ä.), wird dies als eine externe und eigenständige Ontologie begriffen. Dies resultiert aus der Grundannahme, dass Prozesse, Aufgaben etc. eben nicht nur speziell für die Kontextbeschreibung, sondern auch unabhängig davon Verwendung finden (z.B. im Rahmen von Workflowsystemen).

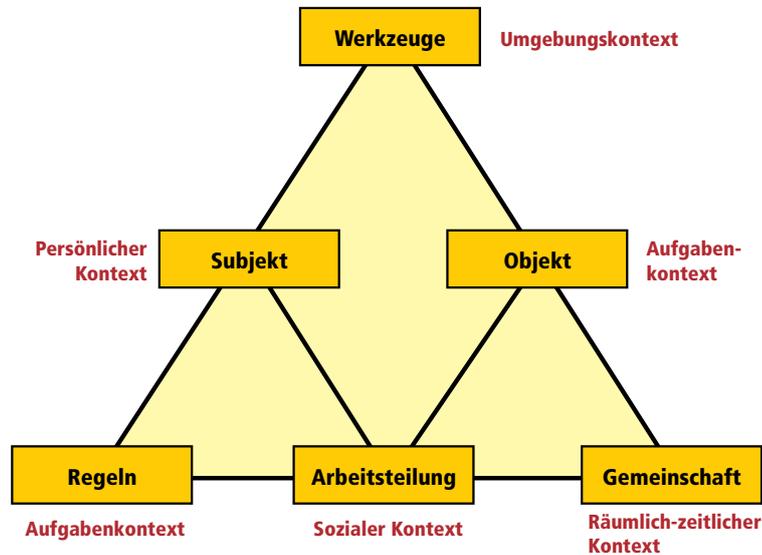


Abbildung 8.3.: Konzepte der Aktivitätstheorie und die Zuordnung von Kontextkategorien nach [KO04]

Damit eignet sich der auf der Aktivitätstheorie basierende Ansatz von KAENAMPORN-PAN als Ausgangspunkt. Dieses setzt auf ein erweitertes Modell der Aktivitätstheorie von [Kuu96] auf (der sog. »kulturreicheren Aktivitätstheorie«). Die ursprüngliche Aktivitätstheorie von VYGOTSKY kennt zur Beschreibung von Aktivitäten drei Konzepte: das *Subjekt*, das ein *Werkzeug* benutzt, um ein *Objekt* zu transformieren. Die kulturreichere Aktivitätstheorie von KUUTTI ergänzt dies (s. Abb. 8.3) um zusätzliche Konzepte. Zentrale Erweiterung ist hierbei die »Gemeinschaft« sowie die Konzepte Regeln und Arbeitsteilung mit Mediationsfunktion. Regeln repräsentieren explizite und implizite Normen einer Gemeinschaft und vermitteln somit zwischen Individuum und Gemeinschaft; Arbeitsteilung bezieht sich auf implizite und explizite Organisationsformen einer Gemeinschaft zur Transformation des Objektes im Rahmen der Aktivität.

KAENAMPORN-PAN identifiziert nun mit Hilfe dieser Theorie eine Kontexttaxonomie (vgl. Abb. 8.3): Regeln und Objekt werden zum Aufgabenkontext, das Subjekt wird zum Persönlichen Kontext, das Werkzeug zum Umgebungs-kontext, die Arbeitsteilung zum Sozialen Kontext und die Gemeinschaft zum räumlich-zeitlichen Kontext. Durch formale Modellierung gelangt er dann zur Modellierung in Abb. 8.4.

Bei genauerer Analyse dieser Modellierungsentscheidung und ihrer Übertragbarkeit auf die Problemstellung dieser Arbeit zeigt sich allerdings, dass die Lösung nicht überall geeignet ist.

case features that are syntactically different, by explaining why they are similar [44,45].

The generic concepts are partly gathered through the use of activity theoretic analysis. These concepts include the six aspects shown in Fig. 3. The top-level taxonomy including the concepts acquired from AT is depicted in Fig. 5. The context model is now primed to model situations and the activities occurring

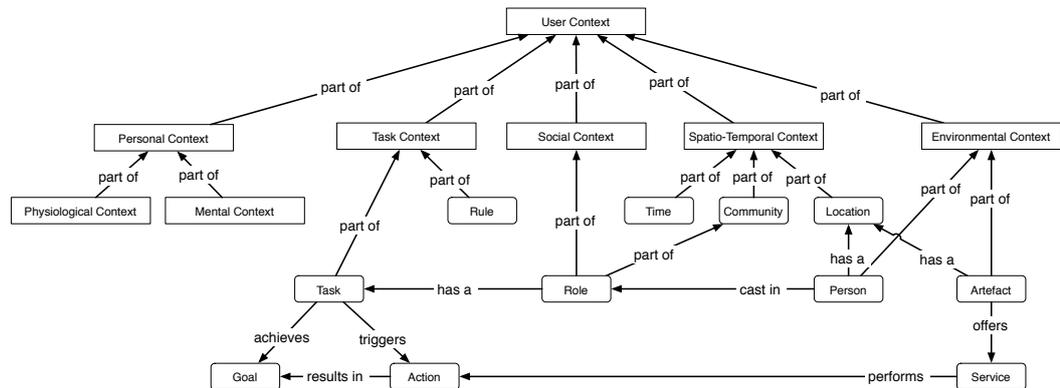


Fig. 5. Populated context structure
Abbildung 8.4.: Kontextmodell nach [KO04]

• Der Umgebungs-kontext (*Werkzeug*) ist sinnvoll, allerdings werden primär mehrere gestützte Arbeiten unterstützt, so dass die Kontextkategorie auf technische Voraussetzungen, per se bestimmt. Inhalt aus sich selbst voll angelegt werden können (Anzeige- und Multimediafähigkeiten, Netzverbindungen etc), so eingeschränkt werden kann. Zur besseren Verständlichkeit wird im folgenden allerdings von »Technischem Kontext« gesprochen.

- Unverändert kann der Persönliche Kontext (*Subjekt*) als Kategorie übernommen werden, wo allerdings eine stärkere Spezialisierung in Richtung Lernen erforderlich wird.
- Die Bezeichnung »Sozialer Kontext« für die *Arbeitsteilung* erscheint irreführend. Stattdessen spielen soziale Aspekte zur Beschreibung der *Gemeinschaft* eine weitaus größere Rolle. Die dort verwendete Bezeichnung »Räumlich-zeitlicher Kontext« wird dem nicht gerecht. Deshalb wird im folgenden die Bezeichnung »Sozialer Kontext« für Aspekte der *Gemeinschaft* verwendet und dabei primär auf die Qualität der Beziehungen zu anderen Personen reduziert.
- Betrachtet man derzeitige Beschreibungsansätze von Organisationsstrukturen (im wesentlichen Ablauf- und Aufbauorganisation), so erkennt man, dass eine Trennung von *Arbeitsteilung*, *Regeln* und *Objekten* von Aktivitäten sich nur schwer vornehmen lässt und diese besser integriert in einem Organisationalen Kontext mit den jeweiligen Querbezügen beschrieben werden.

Damit ergeben sich die folgenden Kontextkategorien oder Ebenen:

- *Technische Ebene.* Die Kontextinformationen auf dieser Ebene umfasst alle Informationen zur Rechnerumgebung.
- *Individuelle Ebene.* Diese Ebene fasst alle Merkmale zusammen, die die Person des Benutzers betreffen, also seine Präferenzen, seine lernbezogenen Eigenschaften (z.B. mit welcher Art von Materialien lernt er am besten), sein Vorwissen etc.
- *Soziale Ebene.* Hier werden die Beziehungen zwischen dem einzelnen Benutzer und seiner menschlichen Umwelt betrachtet.
- *Organisationale Ebene.* Hier wird die organisationale Umgebung abgebildet; hierzu gehören Prozesse, Rollen und Aufgaben.

Hiermit wird im wesentlichen die Grundstruktur der Aktivitätstheorie (allerdings nicht alle Modellierungsentscheidungen von [KO04]) übernommen bis auf die organisationale Ebene, bei der die Untermengen mehrerer Kategorien zusammengefasst werden (wie auch die oben vorgestellten Ergebnisse aus der Kontextmodellierung im Bereich des Wissensmanagement dies tun).

Im Gegensatz zu vielen Benutzermodellierungsansätzen erscheint eine Differenzierung zwischen den Handlungen (*jetzt*) und den Zielen des Benutzers (*Zukunft*) nicht erforderlich, da längerfristige Ziele nur auf den Kompetenzen wirklich sinnvoll sind. Für kurzfristige Ziele (wie sie von Benutzermodellierungsansätzen eingesetzt werden) macht dies für die Granularität, die hier betrachtet wird (Aufgabe, Prozessschritt etc.) nicht wirklich einen Unterschied.

Im folgenden sollen die vier Ebenen noch weiter detailliert werden. Eine wesentliche Frage hierbei ist, welche Elemente auf welcher Detaillierungsstufe in die Ontologie aufgenommen werden sollen. Der Aufnahmeentscheidung müssen drei Kriterien zugrundegelegt werden:

- **Erwartete Nützlichkeit.** Dies gründet auf der Frage: macht es aus methodischer Sicht Sinn, das jeweilige Merkmal zu betrachten? Nur wenn man durch die Kenntnis dieses Kontextmerkmal auch ein besseres Systemverhalten erzielen kann, ist es sinnvoll, dieses auch aufzunehmen. Hierzu gehören Erkenntnis über den Einfluss von bestimmten Kontextaspekten auf Lernsituationen. Um dieses Kriterium zu prüfen wurde auf Lernexperten im Projekt LIP gesetzt sowie im Rahmen einer szenariobasierten formativen Evaluierung die Endbenutzer befragt.
- **Ermittelbarkeit.** Wenn die Kontextontologie kein theoretisches Konstrukt bleiben will, dann müssen Kontextmerkmale auch für die jeweiligen Benutzer ermittelt werden können. Wie im vierten Teil der Arbeit noch darzustellen ist, sind die meisten

Quelle	Merkmale
Workflowsysteme	Prozess, Prozessaktivität
ERP-Systeme	Organisationseinheit, Rolle
integrierte Fachanwendungen	Prozess, Prozessaktivität, Rolle, Aufgabe
Personalmanagement-, speziell Personalentwicklungssysteme	Kompetenzen, Ziele
E-Mail-Client	Aufgabe, soziale Beziehung
Terminkalender	verfügbare Zeit
Office-Anwendungen	Aufgabe
Betriebssystem	Auflösung, Bandbreite, Audioausstattung
Browser	Version, Plugin, Aufgabe
Indirekte Methoden	Merkmale
Ableitung aus Klick-Verhalten	Stress-Zustand
Stresszustand + Tageshistorie aus Terminkalender	Prägnanz
Benutzer-Feedback	Verspieltheit

Abbildung 8.5.: Identifizierte potentielle Kontextquellen zur Überprüfung der Ermittelbarkeit

Kontextmerkmale dabei nicht direkt zugänglich, sondern müssen mittels heuristischer Verfahren gewonnen werden. Da sich dieses Kriterium nicht abstrakt entscheiden lässt, wurden zur Abschätzung der Ermittelbarkeit mögliche Quellen identifiziert (vgl. Abb. 8.5) sowie Algorithmenskizzen angefertigt und im Rahmen eines Rapid-Prototyping durch die Technologiepartner im Projekt LIP umgesetzt.

- **Verfügbarkeit** von Ressourcenmetadaten. Da eine wesentliche Aufgabe die Auswahl passender Lerngelegenheiten darstellt, ist zusätzlich auch danach zu fragen, welche Metadaten für diese Gelegenheiten benötigt werden, um die erwartete Nützlichkeit auch tatsächlich umzusetzen. Hierbei spielen auch Fragen nach dem Aufwand der Metadatenerfassung eine Rolle. Zur Prüfung dieses Kriteriums wurde auf Metadatenstandards (insbesondere IEEE LOM (*Learning Object Metadata*) [Lea02], vgl. Abb. 8.6)⁴ gesetzt, da zu erwarten steht, dass Metadaten kaum über diese Standards hinaus im großen Stile bereitgestellt wird.

Ein sequentielles Abprüfen der einzelnen Kriterien ist allerdings nicht möglich, da die jeweilige Prüfung nicht nur als »Filter« dient, sondern auch zusätzliche Kontextmerkmale vorschlagen kann. Aus diesem Grund ist für die Identifikation der Kontextmerkmale ein iterativer Prozess zu wählen (wie in Abb. 8.7).

Im einzelnen sollen nun dieser iterative Prozess und dessen wesentliche Ergebnisse getrennt nach den unterschiedlichen Kontextkategorien zusammengefasst werden.

⁴LOM ist eine standardisierte Metadatenbeschreibung von Lernressourcen, die einheitliche Kataloge für Lernressourcen ermöglicht. LOM wurde dabei auch in Industrie-Standards wie ADL SCORM übernommen – teilweise mit kleineren Anpassungen.

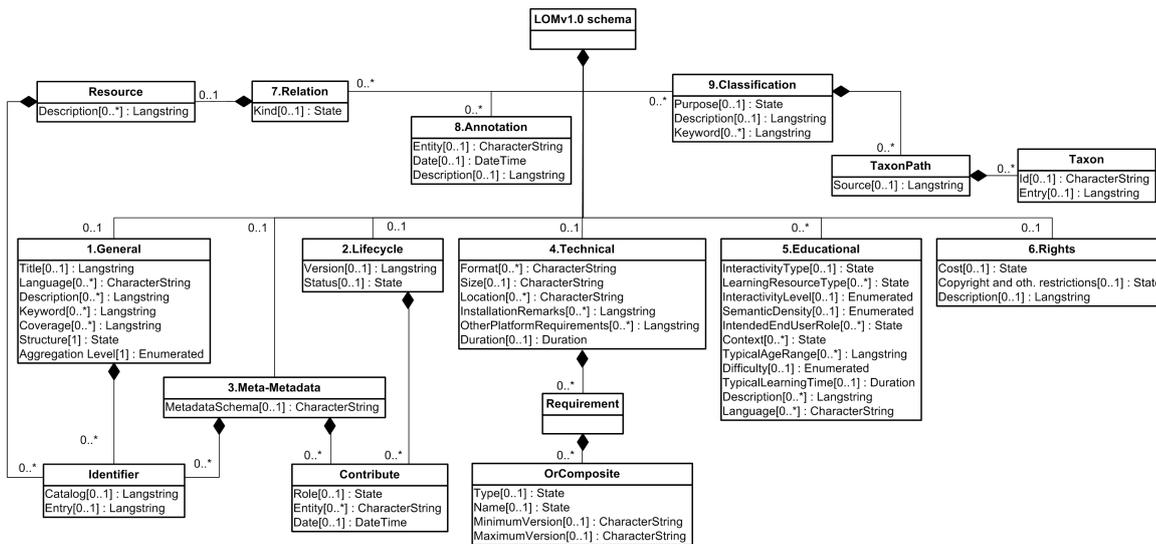


Abbildung 8.6.: Grundstruktur von LOM [Lea02]

Technische Kontextmerkmale

Die technische Ebene ist die am besten verstandene Ebene, da hier die Ermittbarkeit über Standardschnittstellen des Betriebssystems gegeben ist und auch der Nutzen klar ist: man versucht sicherzustellen, dass vom System angebotene Interaktionsformen und Inhalte auch wirklich sinnvoll über das technische Gerät bzw. die Zugangstechnologie zu nutzen sind. Dies wurde bereits bei Standardisierungsinitiativen wie LOM berücksichtigt (der – mit veränderten Kardinalitäten – auch in SCORM benutzt wird), so dass dieses Metadatenschema als Ausgangspunkt genommen werden kann.⁵

Dort findet sich der Abschnitt »technical«, der sich allerdings auf Browser, Betriebssystem und Plugin-Anforderungen beschränkt. Bei der Überprüfung der Nützlichkeit in der Diskussion mit Lernexperten hat sich gezeigt, dass dies alleine nicht ausreicht, um die Angemessenheit von Lernobjekten auf technischer Ebene abzuschätzen. Für die mobile Nutzung ist insbesondere noch die verfügbare Bandbreite hinzuzunehmen, um zu verhindern, dass Videomaterial nicht adäquat dargestellt werden kann (Gegenpart zum LOM-Element *size*). Zudem ist die Verfügbarkeit von Audioausgabe (Headset oder Lautsprecher) nicht überall vorauszusetzen, so dass auch dieses Kriterium hinzugefügt wurde. In vielen praktischen Fällen ist außerdem die Bildschirmauflösung von Bedeutung (diese beiden können

⁵Das teilweise für den Zweck der Beschreibung von Lernerprofilen IEEE PAPI [IEE02] erlaubt zwar deutlich flexiblere Repräsentation von Präferenzen; allerdings existiert für sie dann in den Lernobjektmetadaten kein Gegenstück.

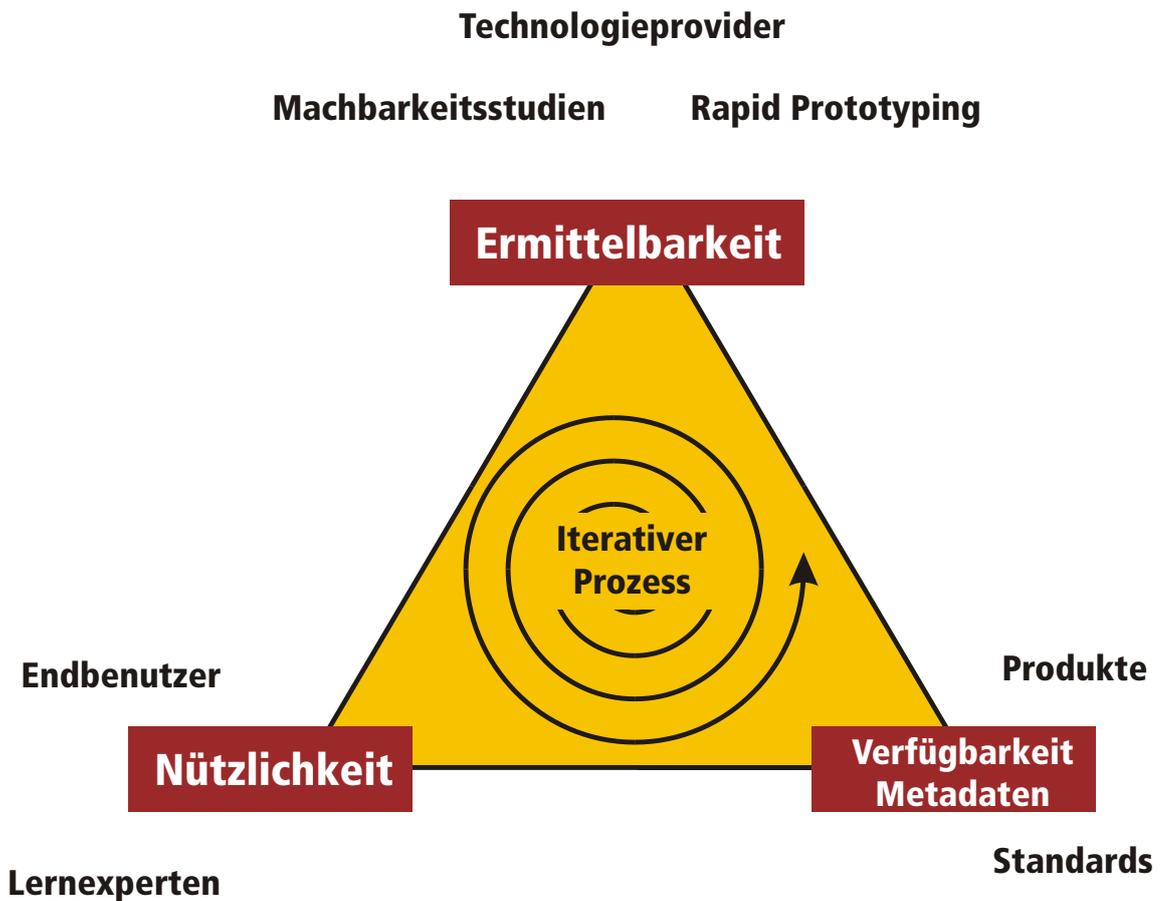


Abbildung 8.7.: Iterativer Abstimmungsprozess zur Erstellung der Kontextontologie im Rahmen des Projektes LIP

als *OtherRequirement* in LOM integriert werden).

Individuelle Kontextmerkmale

Auf der individuellen Ebene spielen die klassischen Merkmale aus dem E-Learning, insbesondere der intelligenten tutoriellen Systeme, eine große Rolle. Ein wichtiger Teil davon (entspricht dem am Anfang des Kapitels gewählten kompetenzorientierten Grundansatz) besteht aus *vorhandenen* und *angestrebten Kompetenzen* (Ziele). Besonders die Bedeutung der Ziele (als Element der Motivation) wird durch die Ergebnisse der Analyse der Informationsrecherche gestützt (vgl. z.B. CHOO in Abschnitt 5.2.3). Damit repräsentiert die individuelle Ebene auch die Dimension der persönlichen Identität [Par00].

Obwohl generell die Bedeutung von Lern- oder Kognitionsstilen betont wird, hat sich zum einen noch keine auch nur annähernd akzeptierte Typologie herausgebildet und zum anderen existieren auch keine Lernobjekte, die auf solche Lernstile zugeschnitten sind. Deshalb werden für die Ontologie nur Lernpräferenzen (die man als Kriterien für Lernstile auffassen kann) betrachtet, die persönliche Vorlieben und Dispositionen dem Lernen gegenüber kennzeichnen. Dabei können sich – im Gegensatz zur klassischen Sichtweise, die von statischen Lernpräferenzen ausgeht – Lernpräferenzen wie andere Kontextmerkmale auch über die Zeit hinweg ändern. Wie bei den technischen Merkmalen ist die Verfügbarkeit entsprechender Metadaten auf Lernobjektseite kritisch. Hier diente der LOM-Metadatenstandard wiederum als Vorlage. Für die Aufnahme in die Kontextontologie erscheinen folgende Lernpräferenzen als sinnvoll:

- *Interaktivitätsgrad*. Dies bezeichnet, wie stark die Aktivität des Lernenden gefragt ist, z.B. im Rahmen von Aufgaben oder aktiver Navigation.
- *Prägnanz* (*conciseness*, in LOM als semantische Dichte bezeichnet). Dies bezeichnet, wie ausführlich und ggf. redundant Inhalte dargestellt werden, z.B. durch Beispiele untermauert, zusätzlich durch Abbildungen illustriert. Für Prägnanz ist meist ein Kompromiss zu finden zwischen Möglichkeit zur schnellen Aufnahmen vs. kognitive Herausforderung.
- *Verspieltheit*. Hier geht es um die Gestaltung der Darstellung: ist sie nüchtern-sachlich mit formeller Ansprache oder hat sie spielerischen Charakter mit lockerer Ansprache und Animationen, die auf den Spieltrieb abzielen. Auch wenn dieses Element nicht in LOM enthalten ist, ist es eine in kommerziellen Angebot vorzufindende Differenzierung, die besonders in interkulturellen Umgebungen von Bedeutung ist (vgl. [DM04]).

In den Diskussion mit den Lernexperten sind über LOM hinaus noch zusätzliche Kontextmerkmale aufgekommen, die die Bereitschaft zu lernen charakterisieren helfen, wozu die *verfügbare Zeit* (korrespondiert mit dem LOM-Element *typical learning time*) und das *Stressniveau* zählen (was eine prominente Rolle z.B. im Modell von WILSON (s. Abschnitt 5.2.1) spielt).

Soziale Kontextmerkmale

Gerade für informelle Lernprozesse ist die soziale Ebene entscheidend, wie in [BS06a] und [BS06b] durch den Autor detaillierter herausgearbeitet wurde. Die Elemente dieser Ebene beziehen sich auf die Dimension der sozialen Identität von Individuen [Par00]. Hierbei steht die Beschreibung der Qualität der Beziehung zu anderen im Vordergrund. Zum einen

sind dies subjektive Bewertungen der Beziehung, zum anderen formelle Beziehungen (Zugehörigkeit zu Gruppen o.ä.).

Für die Charakterisierung sozialer Beziehungen existieren erste Ansätze im FOAF⁶-Umfeld (vgl. [MVB⁺04], [MHM⁺04] und [MG04]). Von einem Standard ist hier allerdings in keinsten Weise zu sprechen, der als Modellierungsgrundlage dienen könnte.

Im Gegensatz zu den anderen Ebenen ist das Kriterium der Verfügbarkeit von Metadaten nicht anwendbar, da es sich um keine Informationsartefakte handelt, die hiermit ausgewählt werden sollen. So bleibt als Kriterium die Nützlichkeit und die Ermittelbarkeit. Besonders die Ermittelbarkeit ist – wenn man nicht auf manuelle Spezifikation durch den Benutzer zurückgreifen will – schwierig, so dass eine minimalistische Modellierung sinnvoll erscheint. Hierzu wird ein Merkmal »knows« definiert, das ausdifferenziert werden kann zu einer organisationalen Unterordnung (»is-subordinate-of«) und einer informellen Qualität (»is-friend-of«). Diese Differenzierung wurde vor dem Hintergrund gewählt, dass Freundschaftsbeziehungen generell förderlich für informelle Lernprozesse sind, während Unterordnungsbeziehungen durchaus problematisch sein können, weil hier Kommunikationsbarrieren zu finden sind.

Organisationale Kontextmerkmale

Im organisationalen Bereich sind im wesentlichen die beiden Kategorien Ablauf- und Aufbauorganisationen zu unterscheiden. Hinzu kommen Aufgaben, die keine feste Verankerung haben, so dass wir folgende Merkmale unter Einbeziehung der oben vorgestellten Ergebnisse zur Kontextmodellierung im Wissensmanagement von GROSS UND KLEMKE und ENGELBACH UND DELP erhalten:

- Organisationseinheit (Aufbauorganisation: Abteilung, Gruppe etc.)
- Geschäftsprozess und -prozessaktivität (Ablauforganisation)
- Rolle
- Aufgabe

Diese Merkmale haben als Wertebereich Konzepte der Organisationsontologie (klassischerweise als Organisationsmodell bezeichnet).

⁶FOAF (»Friend Of A Friend«) ist eine Initiative und RDF-basierte Spezifikation zur Beschreibung dezentraler sozialer Netzwerke, vgl. <http://www.foaf-project.org>

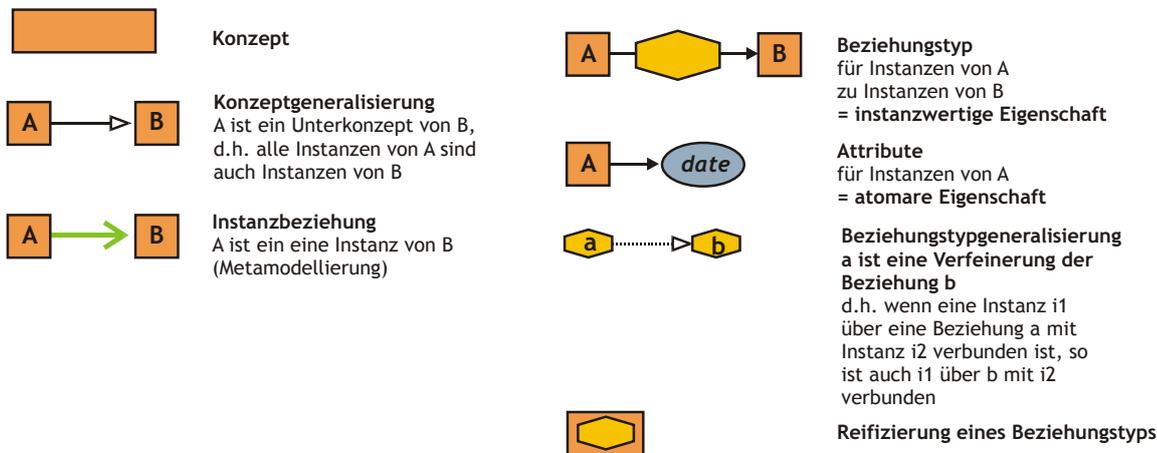


Abbildung 8.8.: Erweiterte ER-Notation für die graphische Darstellung von Ontologien

Graphischer Überblick und Zusammenfassung

Als Zusammenfassung der Ausführungen der letzten Abschnitte findet sich in Abb. 8.9 eine graphische Darstellung der wesentlichen Kontextmerkmale. Für ausgewählte Merkmale wurden zusätzlich die Konzepte angegeben, die die Wertebereiche repräsentieren; die vollständige Kontextontologie findet sich im Anhang A.

Als graphische Notation (die im übrigen auch in den folgenden Kapiteln verwendet wird) wurde hierbei ein erweiterter ER-Diagrammansatz auf der Basis des »Fundamental Modeling Concepts«-Ansatzes [KGT06] und [BL93] ausgewählt, da UML nur schlecht die für eine eigenschaftszentrierte Modellierung erforderliche Darstellung von Generalisierungsbeziehungen auf Eigenschaften unterstützt. Als wesentliche erweiterte Konstrukte werden dabei die Generalisierung von Entitätstypen (im folgenden als »Konzepte« bezeichnet), die Generalisierung von Relationstypen (im folgenden als »Eigenschaften« bezeichnet) und die Reifizierung von Relationen eingesetzt. Weiterhin wurde die Notation um eine Form der Metamodellierung ergänzt, die es erlaubt, Konzepte als Instanzen anderer Konzepte zu verstehen. Die graphischen Konstrukte werden kurz in Abb. 8.8 erläutert.

8.3. Lerngelegenheiten und Kompetenzen

Nachdem nun die Lernenden mit Hilfe der Kontextontologie beschrieben werden können, geht es im nächsten Schritt darum, die Lerngelegenheiten und die Brücken zwischen Lernenden und Lerngelegenheiten (u.a. Kompetenzen) näher zu charakterisieren und formalisieren.

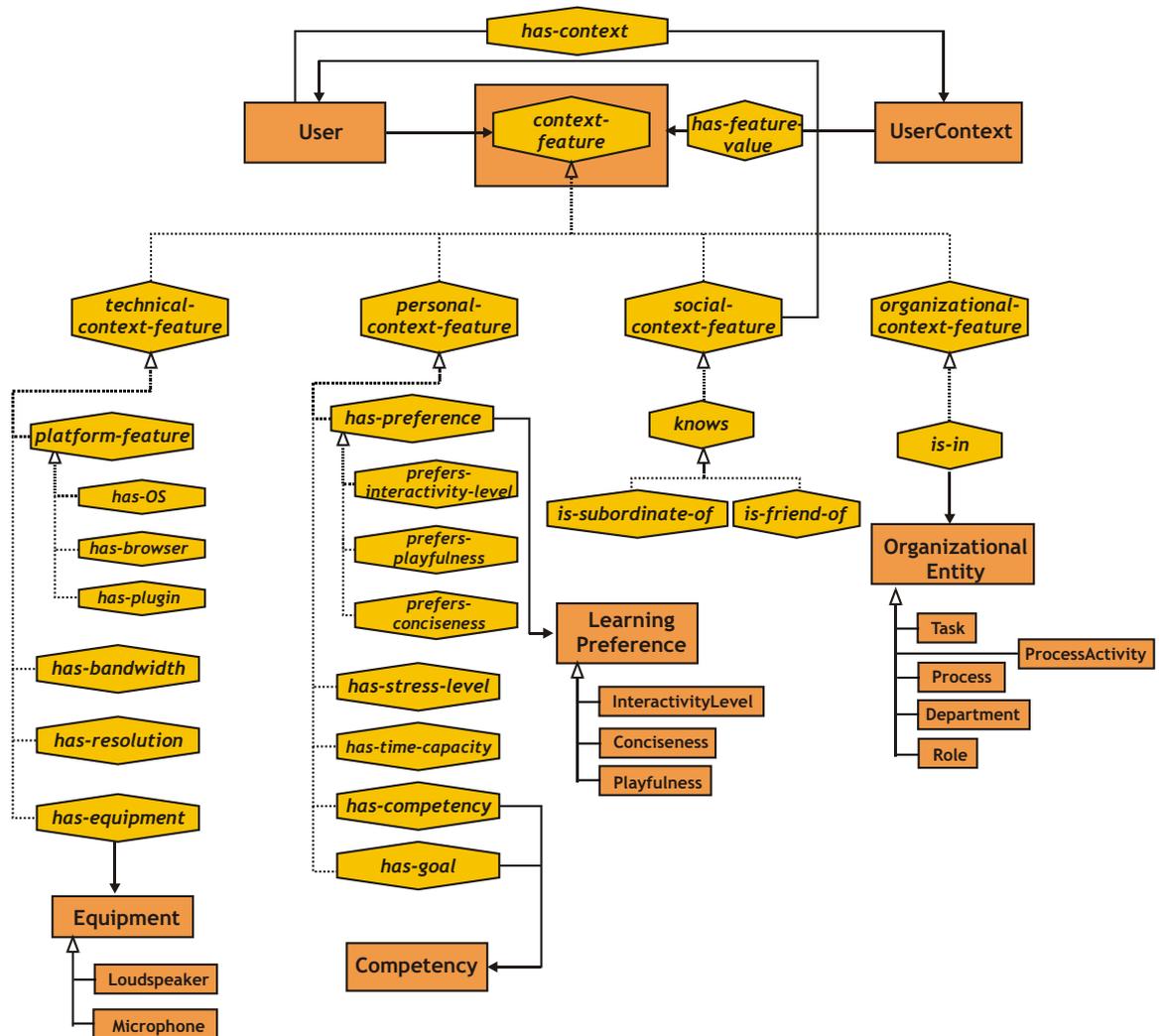


Abbildung 8.9.: Kontextontologie: Überblick

8.3.1. Lerngelegenheit

Lerngelegenheiten (*LearningOpportunity*) werden als Oberbegriff für alles verstanden, was für den Aufbau von Kompetenzen genutzt werden kann. Das schließt sowohl formelle als auch informelle Ressourcen und sowohl Menschen als auch Artefakte mit ein. Durch seine Allgemeinheit kann hierüber relativ wenig ausgesagt werden, insbesondere nicht, welche Kompetenz eine solche Lerngelegenheit vermitteln soll/kann. Allerdings lässt sich eine solche Lerngelegenheit dennoch thematisch charakterisieren. Deshalb wird in die Ontologie das Konzept des Themas (*Topic*) als Verallgemeinerung der Kompetenz eingetragen (vgl. Abb. 8.10); eine ähnliche Konstruktion wird z.B. in [HBV06] als »aspect« bezeichnet.

Wichtige Unterkonzepte der Lerngelegenheit (vgl. Abb. 8.10) sind:

- **Mitarbeiter** (*Employee*). Bei Mitarbeitern wird das Verfügen über Kompetenzen als semantische Spezialisierung der thematischen Beziehung repräsentiert.
- **Informationsartefakte**. Dabei handelt es sich um elektronische Artefakte, die von Kommunikationsfragmenten über Dokumente bis hin zu großen Einheiten wie Büchern reichen können. Neben einem oder mehreren Themen haben sie auch kontextuelle Metainformationen, die sie mit ihrem organisationalen Umfeld in Beziehung setzen können. Die zwei wichtigsten Arten von kontextueller Metainformation sind hierbei der Entstehungskontext (wann, von wem und in welcher Situation wurden sie erstellt) und Nutzungskontexte (wann, von wem und in welcher Situation wurden sie genutzt). Informationsartefakte können zudem einen recht unterschiedlichen Reifegrad (vgl. Kapitel über den Wissensreifungsprozess) haben, was sich ableitet aus dem Grad der Vermittelbarkeit außerhalb des eigentlichen Entstehungskontextes und auch mit dem Grad der didaktischen Gestaltung gleichgesetzt werden kann.
- **Lehrentitäten** (*InstructionalEntity*). Im Gegensatz zu den anderen Lerngelegenheiten sind diese speziell für das Lernen entwickelt und darauf hin optimiert worden. Dementsprechend lassen sich hier auch höhere Ansprüche stellen. Hierzu gehören insbesondere zwei Blöcke:
 - **Ziele und Voraussetzungen** (*has-objective* bzw. *has-prequisite*). Dieser Block ist für die vorliegende Arbeit der wichtigste und legt ein im folgenden Abschnitt noch näher erläutertes Kompetenzmodell zugrunde. Es wird beschrieben, welche Kompetenzen im Falle der »erfolgreichen« Bewältigung der mit der Lehrentität verknüpften Lernaktivität im Normalfall erworben werden und welche Kompetenzen zum Verständnis und zum Kompetenzerwerb notwendig sind.
 - **Abhängigkeiten** (*depends-on*). Nicht alle Zusammenhänge lassen sich vernünftig in isolierten Lehrentitäten abhandeln, so dass auch direkte Abhängigkeiten zwischen solchen Entitäten bestehen können. Diese können dabei Empfeh-

lungscharakter haben (»man sollte zunächst sich dieses Objekt anschauen, bevor man das andere bearbeitet« – wenn man beide benötigt) oder aber auch echten Abhängigkeitscharakter (z.B. wenn auf ein Beispiel Bezug genommen wird, das in einem anderen Lernobjekt eingeführt wird).

8.3.2. Informationsontologie

Das Konzept *ExplicitLearningResource* fasst Informationsartefakte und Lehrentitäten zusammen (vgl. Abb. 8.10). Für ihre Beschreibung existiert eine **Informationsontologie** (was als Synonym für ein klassisches Metadatenschema zu betrachten ist).

Ein für die Arbeit wichtiger Spezialfall sind **Lernobjekte** (*LearningObject*), die Informationsartefakte (*InformationArtefact*) mit einem hinreichenden Grad didaktischer Gestaltung darstellen (also gleichzeitig auch Lehrentitäten sind (*InstructionalEntity*)), die überdies als weithin in sich didaktisch abgeschlossen betrachtet werden können [Wil00]. Für diese existieren auch bereits zahlreiche Ansätze für entsprechende Lernobjektmetadaten. Hierbei standen jedoch stärker das klassische Interaktionsmodell (Suche durch Lehrende bzw. Lernende) sowie reine Verwaltungsfragen (wie Rechtemanagement, Lebenszyklus u.ä.) im Vordergrund, so dass die Ansätze nicht unmittelbar anwendbar sind. Als Ausgangspunkt für die Entwicklung des Metadatenmodells eignet sich, wie bereits oben im Rahmen der Kontextontologie ausgeführt, der De-Facto-Standard IEEE LOM (Learning Object Metadata) [Lea02], der auch – mit veränderten Kardinalitäten – die Grundlage für ADL SCORM bildet. Hierbei sind vor allem die folgenden beiden Metadatenblöcke relevant:

- **Lerncharakteristika.** Lernende unterscheiden sich bzgl. der von Ihnen bevorzugten Lernmaterialien (was durchaus auch situationsabhängig sein kann. Sie lassen sich durch charakteristische Eigenschaften beschreiben wie z.B. semantische Dichte oder Interaktionsgrad. Diese werden bereits im IEEE LOM Standard abgedeckt.
- **Technische Charakteristika.** Dieser Bereich umfasst Metadaten, die Anforderungen an die technische Umgebung des Lernenden beschreiben. Hierzu gehören installierte Software/Plugins, aber auch Hardware-Ausstattung wie z.B. Lautsprecher oder die benötigte Bandbreite.

8.3.3. Kompetenzontologie

Kompetenzen und Kompetenzkatalog

Die Modellierung von Kompetenzen ist inzwischen eine gängige Praxis; allerdings liegt dem in den seltensten Fällen ein formales konzeptuelles Modell, oft nicht einmal eine

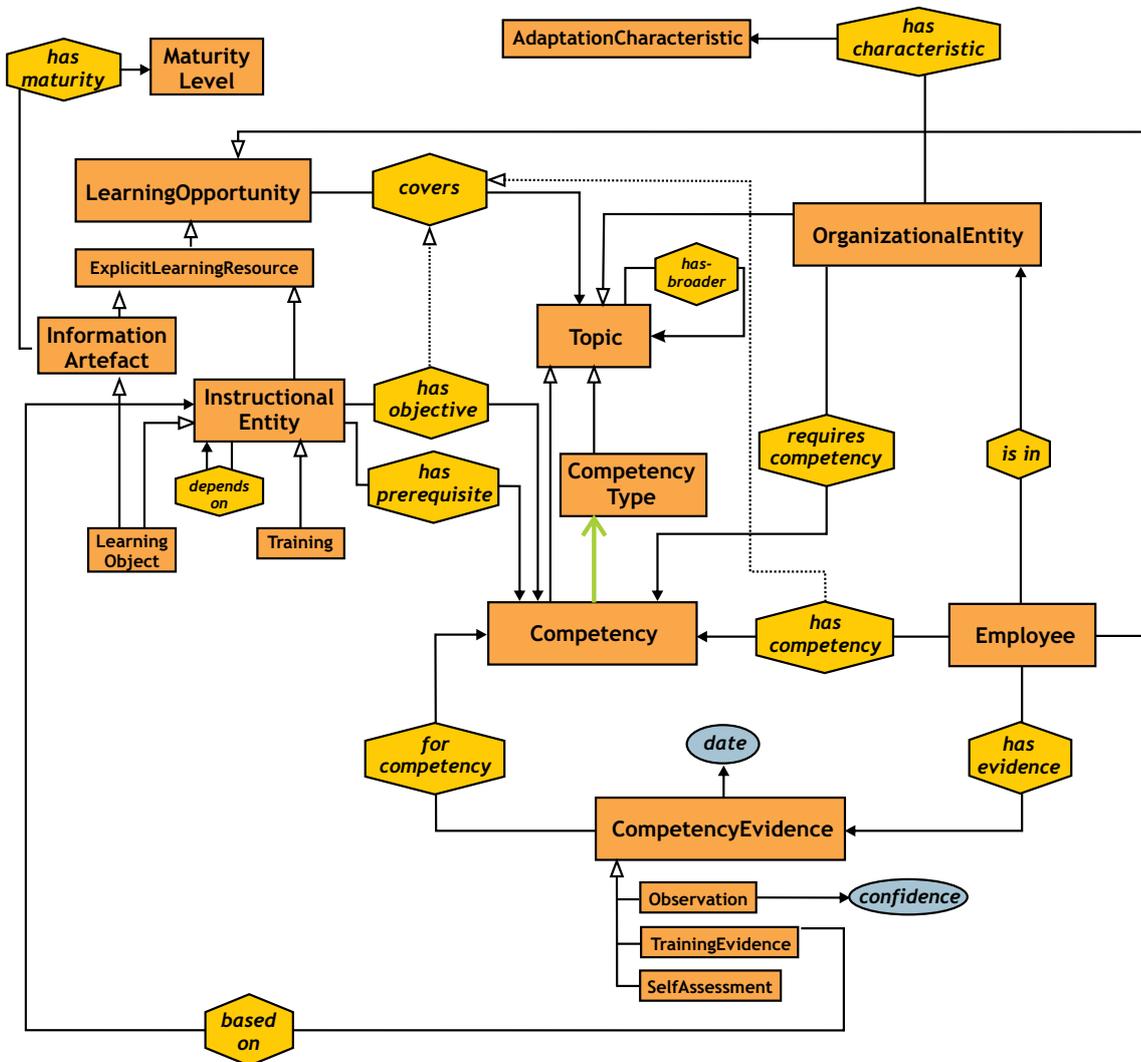


Abbildung 8.10.: Referenzontologie: Gesamtüberblick

Kompetenzdefinition zugrunde. Werden Kompetenzen primär als Strukturierungsinstrument für weiterhin »manuelle« Entscheidungen benutzt, so ist dies auch nicht unbedingt erforderlich, und es geht eher darum, die Modelle durch pragmatische Vorgehensweise möglichst kompakt und damit beherrschbar zu halten (vgl. z.B. [Coo00]). Für einen automatisierten Umgang ist allerdings eine Formalisierung unumgänglich.

Hier existieren bislang nur wenige Ansätze. Im Bereich der Standardsierung liegt mit IEEE RCD (Reusable Competency Definitions) [IEE07] nur ein rudimentärer Ansatz vor, der für einzelne Kompetenzen eine austauschbare XML-Repräsentation festlegt, die im wesentlichen aus einem eindeutigen Identifikator, einem Titel, einer Beschreibung und nicht weiter spezifizierten Metadaten bestehen. Ähnlich ist auch der Industriestandard HR-XML [All01], der zwar durch die Möglichkeit zur Verwendung von externen Taxonomien Brücken zu semantischen Ansätzen schlagen kann, aber in seiner Spezifikation nur syntaktisch von IEEE RCD abweicht.

Auf der konzeptuellen Ebene existieren zwei Ansätze. Mit dem Fokus auf logischen Schlussfolgerungen auf Kompetenzaussagen wurde in [ZA05] eine generische Kompetenzontologie entwickelt; hier lassen sich die Grundstrukturen für die Repräsentation des Kompetenzerwerbs verwenden. BIESALSKI hat für die Anwendung ontologiebasierter Verfahren für die Personalentwicklung ein ontologiebasiertes Kompetenzmodell entwickelt [Bie06], das bis auf einzelne im folgenden hervorgehobene Punkte im wesentlichen in eine Untermenge der hier präsentierten Domänenontologie abgebildet werden kann.

Zunächst soll einmal der Kompetenzbegriff näher definiert werden: Während Kompetenz (im Singular) sowohl die Bedeutung der Gesamtheit der Fähigkeiten, Fertigkeiten und Kenntnisse in Bezug auf einen bestimmten Handlungskomplex (also letztlich die »Handlungskompetenz«) als auch die Bedeutung einer von anderen abgrenzbaren Einzelkompetenz besitzt (vgl. [Kun05]), soll Kompetenz für diese Arbeit pragmatisch wie folgt definiert werden:

Definition 8.3 (Kompetenz) *Eine Kompetenz soll verstanden werden als (1) ein Bündel von Fähigkeiten, Fertigkeiten und Kenntnissen, die (2) relevant für die berufliche Leistung sind, (3) hinreichend messbar/beobachtbar sind und (4) sich entwickeln lassen. (vgl. [Kun05], ähnlich [Kir04] und HR-XML [All04]).*

Dieser Kompetenzbegriff vereinfacht dabei das Konstrukt »Kompetenz« gegenüber anderen Ansätzen zur Definition des Begriffes im Bereich der (Berufs-)Pädagogik und Psychologie, um es zu operationalisieren und somit für eine maschinenverarbeitbare Semantik zugänglich zu machen. Die in der Definition enthalten Kriterien sind dabei als ein »kleinster gemeinsamer Nenner« der unterschiedlichen Kompetenzbegriffe zu verstehen, die unterschiedliche Kompetenzmodelle ermöglichen (z.B. auf der Basis von ERPENBECK & ROSENSTIEL [ER03]). Mit anderen Worten versucht die hier vorgestellte Teilontologie

einen allgemeinverbindlichen Rahmen festzulegen, innerhalb dessen die Freiheiten bestehen, wie man beispielsweise Kompetenzen definiert und wie man sie misst bzw. messbar macht. Dies ist auch notwendig angesichts der Tatsache, dass kein allgemein akzeptiertes Kompetenzmodell absehbar ist.

In der obigen Kompetenzdefinition ist auch implizit enthalten, dass Kompetenzen im Normalfall domänenspezifisch sind und nicht auf generische Kompetenzen (wie z.B. Schlüsselkompetenzen bzw. -qualifikationen, [Mer74]) abgezielt wird. Generische Kompetenzen machen aus bildungspolitischer Sicht Sinn, um Grobziele vorzugeben. Für ein konkretes Eingreifen, insbesondere im Rahmen des Lernen bei Bedarfs, ist eine feinere Granularität erforderlich. Ähnlich sieht das auch [CLMM07], wobei dort zusätzliche Modellierungsannahmen getroffen werden, nämlich eine Differenzierung zwischen allgemeinen, generischen Kompetenzen und einem Kontext, in dem sich diese manifestieren.

Üblicherweise sind Kompetenzen keine scharfen Konzepte, die entweder vorhanden sind oder nicht, sondern müssen als abgestufte Instanzen eines Kompetenztyps gesehen werden.⁷ Hierzu werden ordinale **Kompetenzskalen** benutzt (vgl. auch [Bie06]). Eine bekannte und häufig verwendete Skala ist die Skala nach DREYFUS & DREYFUS [DD86] (vgl. Abschnitt 2.6) mit den Skalenstufen Neuling, Anfänger, Fortgeschrittener, Kompetenter, Experte. Ein anderes Beispiel für eine Kompetenzskala ist die für die europäischen Referenzniveaus für Sprachkenntnisse (CEFR, [Cou01]), die aus den Niveaus A1, A2, B1, B2, C1 und C2 bestehen. Wie man an den Beispielen bereits erkennen kann, sind Skalen meist nicht universell auf alle Kompetenztypen anwendbar, sondern müssen einzelnen Kompetenztypen zugeordnet werden.

Als Konsequenz für die Modellierung (vgl. Abb. 8.11) ergibt sich folgendes: Kompetenztypen werden als Unterkonzepte des Basiskonzeptes *Competency* modelliert; ihre Instanzen sind Kompetenzen, denen ein bestimmter Skalenwert (*CompetencyLevel*) zugeordnet wird. Für die Zuordnung einer Skala (*CompetencyScale*) werden Kompetenzen als Instanzen von Kompetenztypen (*CompetencyType*) begriffen. So kann ein Kompetenztyp für Sprachkompetenzen definiert werden, dem die CEFR-Skala zugewiesen wird.

Die Summe von Kompetenzen, der Festlegung von Kompetenzskalen und Kompetenzausprägungen (Instanzen) können zu einem **Kompetenzkatalog** zusammengefasst werden, wie in der Übersicht gezeigt.

Für umfangreiche Kompetenzkataloge erscheinen Beziehungen zwischen Kompetenzen sinnvoll, auch wenn in der Praxis derzeit nur informelle hierarchische Beziehungen existie-

⁷An dieser Stelle ist auf einen Unterschied zu den Ansätzen in [Bie06] und in [ZA05] hinzuweisen. Dort wird der Begriff »Kompetenz« für »Kompetenztyp« und »Kompetenz-Instanz« bzw. »Kompetenzausprägung« für »Kompetenz« benutzt. Die Hauptmotivation für die Modellierungsentscheidung dieser Arbeit war die natürliche Repräsentation von Aussagen wie »ein Mitarbeiter hat die Kompetenz XY«. Außerdem zeigen curriculare Ansätze auch oft, dass es sinnvoller ist, ein bestimmtes Niveau eines Kompetenztyps als eigenständige Kompetenz zu sehen.

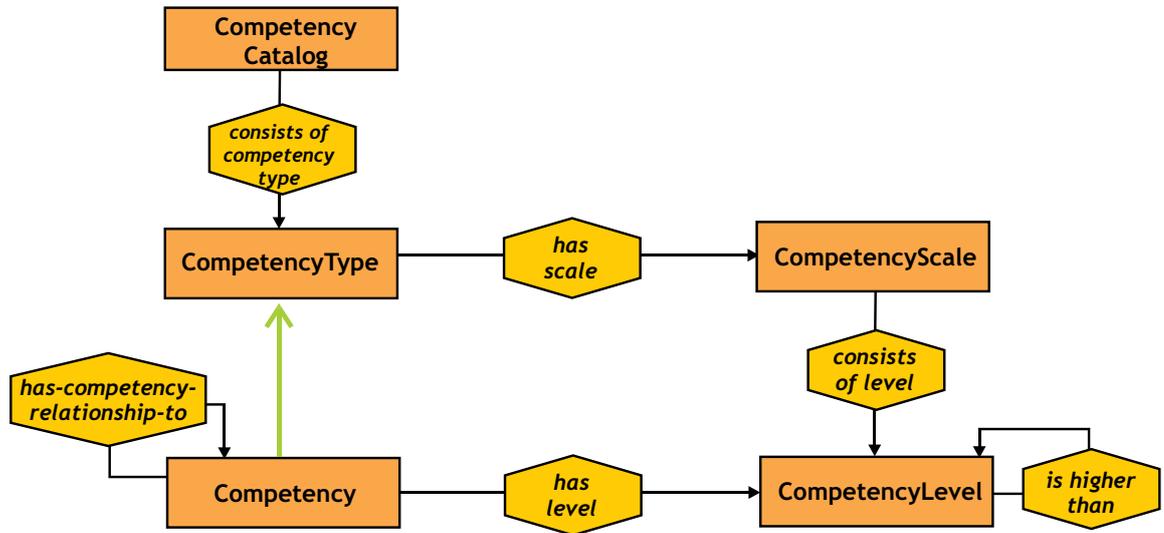


Abbildung 8.11.: Modellierung von Kompetenzen in der Referenzontologie (Detailansicht zu Abb. 8.10)

ren [SKB06]. Solche Beziehungen können helfen, um das Problem der unterschiedlichen Abstraktionsebenen beim Umgang mit Kompetenzen zu bewältigen. So kann es beispielsweise sein, dass an der einen Stelle ein Mitarbeiter gesucht werden, der sich mit Projektmanagement sehr gut auskennt, während bei der Auswahl von Lerngelegenheiten zur Projektmanagementkompetenz feiner differenziert werden muss, z.B. Beherrschen von »Microsoft Project«. Die Auswahl solcher sinnvoller Beziehungen und insbesondere die Definition der Semantik im Hinblick auf kompetenzbasierte Operationen ist allerdings ohne die Betrachtung der Operationen, die diese nutzen, selbst kaum sinnvoll zu bewerkstelligen, so dass wir die Ausdifferenzierung von Kompetenzbeziehungen auf die eingehende Untersuchung in Kapitel 10 verschieben, wo es um die Implementierung wiederverwendbarer kompetenzbasierter Dienste geht.

Kompetenzerwerb

Im vorangegangenen Abschnitt haben wir die modellhafte Sicht eingeführt, dass Lehrentitäten über Lernziele verfügen und dies so interpretiert, dass bei erfolgreicher Bewältigung ein Kompetenzerwerb unterstellt wird. Hierbei sollte klar sein, dass diese Betrachtung die realen Verhältnisse sehr stark vereinfacht, da zum einen das Kompetenz-Performanz-Problem ([Kor97], [HCWA03], [Hoc03], [LA03a], [LA03b]) (d.h. die beobachtbare Performanz läßt nur indirekte Schlüsse auf die vorhandene, aber nicht beobachtbare Kompetenz eines Mitarbeiters zu) ignoriert und zum anderen unterstellt wird, dass von einem »Nor-

malfall« tatsächlich ausgegangen werden kann, was bei konstruktivistischer Sichtweise durchaus fragwürdig ist.

Um dieser Vereinfachungssituation Rechnung zu tragen, wird in Anlehnung an HRXML [All01] das Konzept des Kompetenzindizes (*CompetencyEvidence*) eingeführt und die Kompetenzrelation (*has-competency*) als abgeleitete Beziehung betrachtet, die eine (unvollkommene) Schlussfolgerung darstellt (vgl. Abb. 8.10). Als Kompetenzindiz kann neben dem erfolgreichen Abschluss von Präsenztrainings oder Lernobjekten (überprüft z.B. durch Lernerfolgskontrollen) auch die Selbstbewertung oder die Fremdbewertung durch Vorgesetzte dienen. Sie alle können nie zweifelsfrei das Vorhandensein einer Kompetenz feststellen, sondern nur Indizien bilden, die in einer Zusammenschau das Vorhandensein einer Kompetenz plausibel machen.

8.3.4. Organisationsontologie

Wie in Kapitel 2 betont spielt beim arbeitsbegleitenden Lernen die Steuerung im Sinne der Unternehmensziele eine Rolle. Kurzfristig bedeutet dies, dass Mitarbeiter über alle für die optimale Bewältigung der Aufgaben in den jeweiligen Geschäftsprozessen erforderlichen Kompetenzen verfügen; mittel- und langfristig heißt es aber auch, dass die Mitarbeiter auf mehr oder minder antizipierbare Veränderungen vorbereitet sind, so dass sie mit veränderten Prozessen zurechtkommen und aktiv die Prozesse selbst besser gestalten können. Dieses abstrakte Ziel wird im Rahmen von Kompetenz- und Skills-Management-Ansätzen durch **Anforderungsprofile** operationalisiert (vgl. [Kun05] bzw. [KS06]).

Diese legen fest, welche Kompetenzen erforderlich sind und werden üblicherweise an Elemente der Aufbauorganisation (z.B. Abteilungen oder Rollen) bzw. Ablauforganisation (z.B. Prozess(aktivitäten) oder Aufgaben) geknüpft. Diese werden in der Modellierung zu Organisationsentitäten (*OrganizationalEntity*) abstrahiert. In diesen Profilen lässt sich unterscheiden zwischen unbedingt erforderlichen Kompetenzen (*absolutely-requires-competency*) und erwünschten Kompetenzen (*desires-competency*, vgl. Abb. 8.12 als Detailausschnitt zu Abb. 8.10).

8.3.5. Bezug zu anderen Modellen im Unternehmen

Die Forderung nach Integration in das Unternehmen (vgl. Anforderung AN1) ist insbesondere für Modelle kritisch, da ihre Aufstellung teuer ist. Hier ist also weitestgehende Mehrfachverwendung Pflicht. In diesem abschließenden Abschnitt soll kurz umrissen werden, wie sich die Domänenontologie hierfür eignet (vgl. auch Abb. 8.13):

- Die **Organisationsontologie** ist die Schnittstelle zum Geschäftsprozessmanagement und zur Aufbauorganisation. Da für unsere Domänenontologie nur die Identität sol-

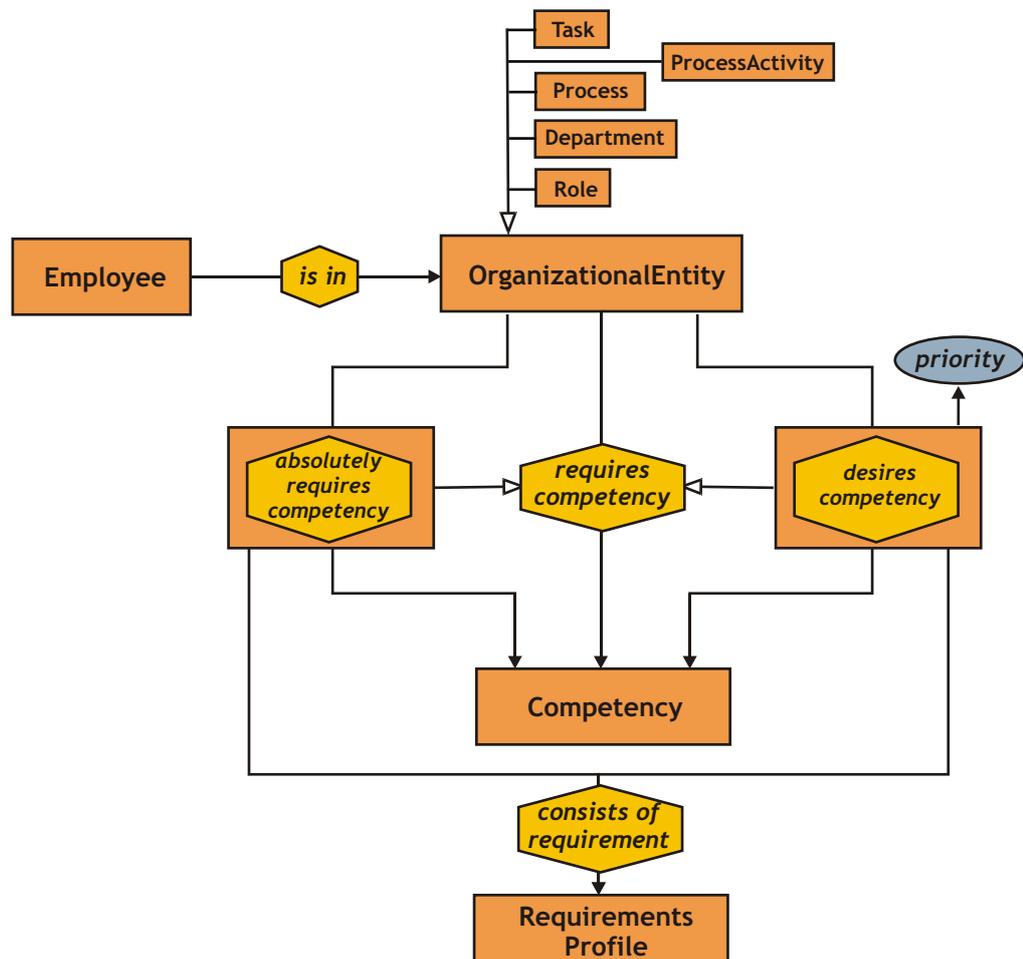


Abbildung 8.12.: Referenzontologie: Modellierung der Anforderungsprofile (Detailansicht zu Abb. 8.10)

cher Entitäten wie Prozesse, Rollen oder Abteilungen notwendig sind, ist die automatisierte Abbildung existierender Modelle kein Problem.

- Die **Kompetenzontologie** ist die Schnittstelle zur Personalentwicklung bzw. zum (individuumszentrierten) Kompetenzmanagement (vgl. auch [SKB06]). Durch die unmittelbare Integration in die Arbeitsprozesse muss die Kompetenzontologie notwendigerweise feingranularer sein. Hier kann aber dann davon ausgegangen werden, dass dies eine Verfeinerung des Kompetenzkataloges aus dem Kompetenzmanagement ist, wenn vorausgesetzt wird, dass dies formal modelliert ist. Ähnliches gilt für **Anforderungsprofile**, die ebenfalls feingranularer gehalten werden müssen (z.B.

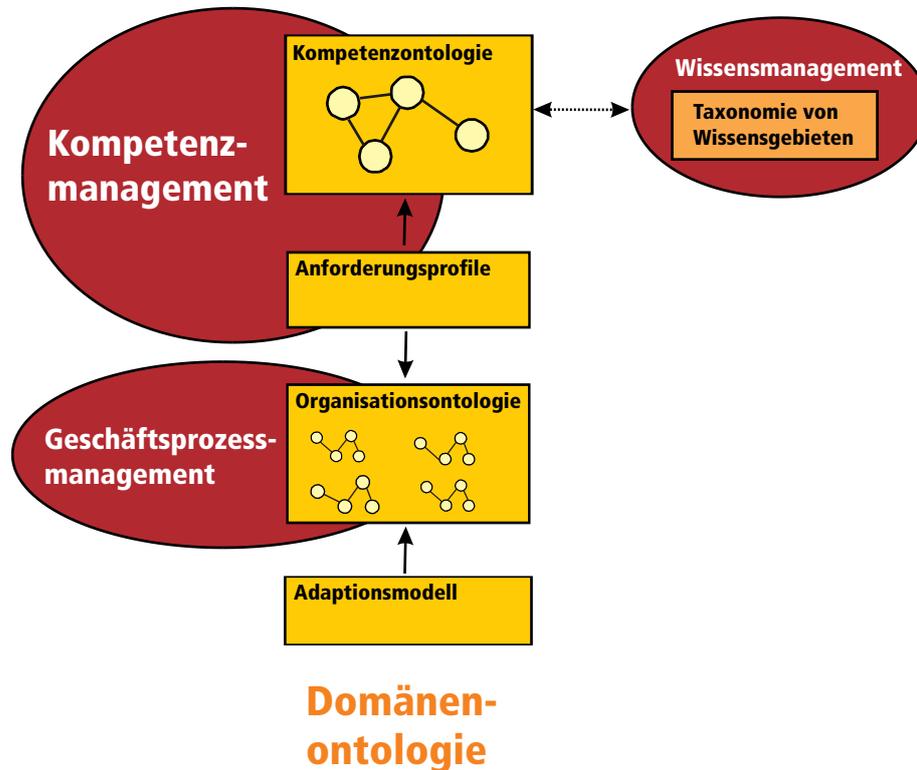


Abbildung 8.13.: Das Domänenmodell und seine Integrationsfunktion mit den unterschiedlichen Gebieten des arbeitsbegleitenden Lernens

nicht nur auf Rollen, sondern auch auf Aufgaben und Prozessaktivitäten).

- Indem wir Kompetenzen als spezifische *Topics* aufgefasst haben, sind wir auch in der Lage, übliche Taxonomien auf Wissensgebieten für das **Wissensmanagement** einzubinden.

Darüber hinaus können selbstverständlich existierende Repositorien für E-Learning-Inhalte (Lernobjekte) eingebunden werden; hier sind allenfalls zusätzliche Metadaten notwendig, sofern noch keine kompetenzorientierte Beschreibung vorliegt.

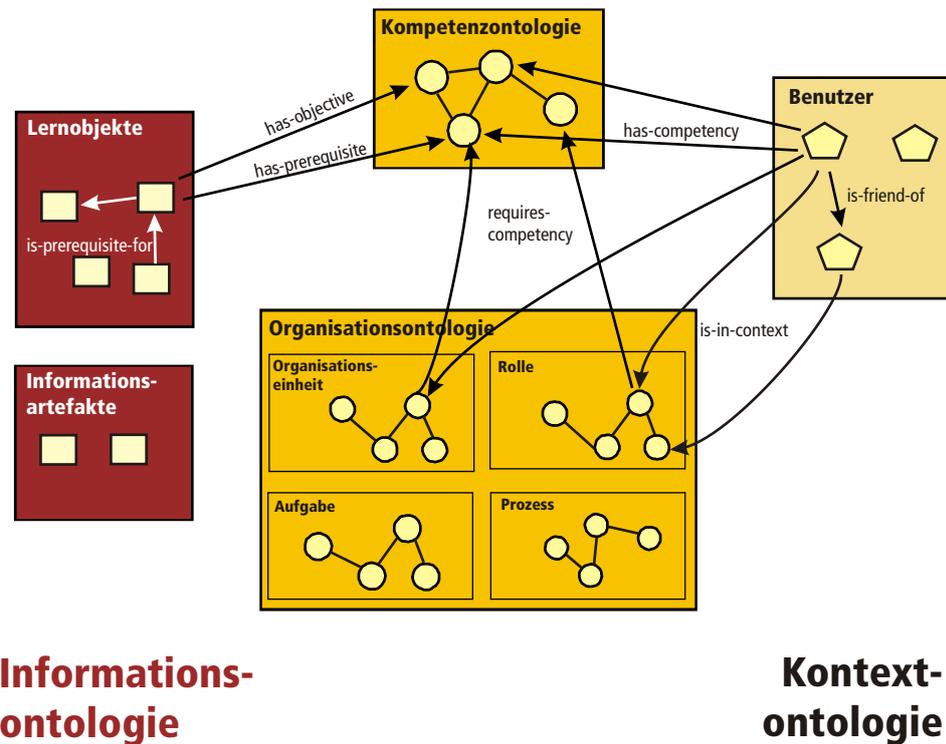


Abbildung 8.14.: Vereinfachte Grundstruktur der Domänenontologie aus Abb. 8.10

8.4. Fazit

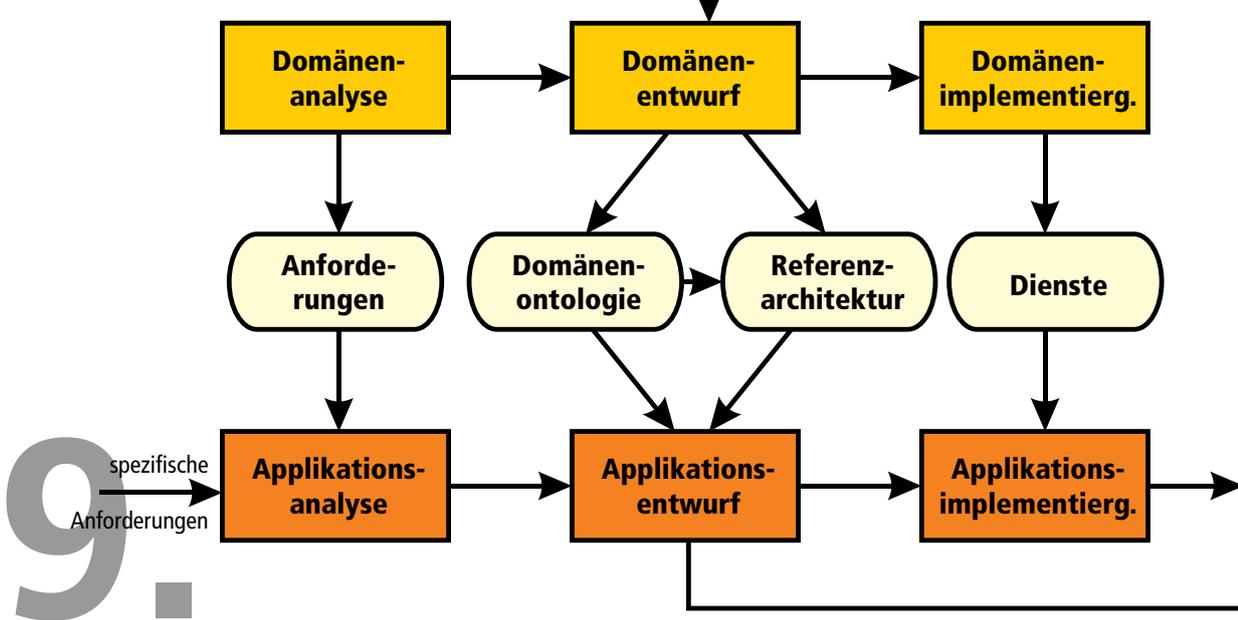
In diesem Kapitel wurde die Domänenontologie entwickelt, die die in Abb. 8.14 dargestellte Grundstruktur besitzt.⁸ Auf der rechten Seite befinden sich die menschlichen Benutzer, die durch die Kontextontologie beschrieben werden. Auf der linken Seite finden sich die Informationsartefakte (mit ihrem wichtigsten Spezialfall der Lernobjekte), deren Beschreibung als Informationsontologie bezeichnet werden soll. Beide Seiten verbinden eine Kompetenzontologie und eine Organisationsontologie, die durch Anforderungsbeziehungen miteinander verknüpft sind.

Mit diesem Kapitel wurde die Grundlage geschaffen für einen ontologiezentrierten Architekturentwurf, in dem für die Unterstützung des situationsgestützten Lernens eine Konzeptualisierung in Form einer Domänenontologie erarbeitet wurde. Diese ist konform mit den Anforderungen in Abschnitt 7.1 und 7.2:

⁸Eine Umsetzung der Ontologie in OWL-DL, wie sie für die weitere Implementierung verwendet wurde, findet sich in Anhang A.

- Für die Umsetzung der Anforderungen AF1-4 ist mit dem kompetenzorientierten Grundansatz die Grundlage geschaffen, um Lernbedarf und verfügbare Lerngelegenheiten automatisiert ermitteln zu können.
- Durch die zentrale Abstraktion der Lerngelegenheit, die Lernobjekte, allgemeine Informationsartefakte und Personen umfasst, wird die Modellierung auch den Anforderungen AFL1-3 gerecht.
- Die Kontextsubontologie deckt die Anforderungen AFK1-6 ab; die dort aufgeführten Kontextaspekte wurden im Rahmen eines iterativen Prozesses konkretisiert und als Kontextmerkmale modelliert.
- Wie in Abschnitt 8.3.5 dargelegt, ist die Integration in die Unternehmensumgebungen aus Anforderung AN1 erreicht.

Mit diesem Stand der Domänenontologie kann nun der weitere Architekturentwurf vorangetrieben werden, in dem es hauptsächlich darum gehen wird, die benötigte Funktionalität sinnvoll in Dienste aufzuteilen und diese entsprechend anzuordnen. Wenn es dann um die Realisierung dieser Dienste durch die Entwicklung von Algorithmen geht, wird diese Ontologie noch weiter verfeinert. Hierzu ist allerdings eine vertiefte dienstspezifische Problemanalyse notwendig, um eine adäquate Modellierung vornehmen zu können.



Referenzarchitektur für situationsbewusste lernunterstützende Systeme

Auf der Basis der entwickelten Domänenontologie (als Ergebnis der Domänenanalyse im *Domain Engineering*-Ansatz) muss im nächsten Schritt die Architektur (Schritt 2: Domänenentwurf) entwickelt werden. Während der ontologiezentrierte Architekturstil (wie in Abschnitt 7.4 ausgeführt) zusammen mit der Domänenontologie insbesondere den nicht-funktionalen Anforderungen gerecht wird, geht es nun um Antworten auf die folgenden Fragen:

- Welche Dienste werden benötigt, um die Funktionalität zu realisieren?
- Wie werden diese Dienste in einer Gesamtarchitektur angeordnet, um die Komplexität beherrschbar zu halten?

Im folgenden wenden wir für den Architekturentwurf einen Top-Down-Ansatz an, d.h. zunächst wird die Grobstruktur der Architektur erarbeitet, bevor in ihr die einzelnen Dienste identifiziert werden. Zur Erarbeitung der Grobstruktur werden im folgenden Abschnitt zunächst existierende Architekturansätze analysiert, um bereits bewährte Architekturkonzepte abzuleiten.

9.1. Existierende Architekturen

Da das situationsgesteuerte Lernen nicht einer klassischen »Disziplin« zuzuordnen ist, sind bei der Analyse existierender Ansätze mehrere Bereiche zu betrachten. Zum einen

sind das technologieorientierte Architekturen für kontextbewusste Anwendungen, zum anderen sind es anwendungsorientierte Ansätze im Bereich des adaptiven E-Learning und E-Learning-Plattformen im allgemeinen sowie Wissensmanagementarchitekturen. Im folgenden sollen vielversprechende Vertreter der angesprochenen Bereiche analysiert werden.

9.1.1. Kontextbewusste Anwendungen

HENRICKSEN UND INDULSKA [HI06] schlagen eine Systemarchitektur für kontextbewusste Anwendungen (die vorangegangene Architekturansätze verschmilzt) mit den folgenden Ebenen vor, die auch für unsere Zwecke als Ausgangspunkt sehr brauchbar erscheint (vgl. Abb. 9.1):¹

- **Anwendung** (*application layer*). Dies stellt die eigentliche kontextbewusste Anwendung dar.
- **Anpassung** (*adaptation layer*). Auf dieser Ebene findet die Anpassung des Anwendungsverhaltens an den aktuellen Kontext statt.
- **Anfrage** (*query layer*). Diese Ebene bearbeitet deskriptive Anfragen an die verfügbaren Kontextinformationen.
- **Kontextmanagement** (*context management layer*). Auf dieser Ebene werden die Kontextinformationen gespeichert und konsistent gehalten.
- **Kontextaufnahme** (*context reception layer*). Dies bezeichnet das Einbringen von gemessenen Kontextinformation in das Kontextmanagementsystem.
- **Kontextmessung** (*context gathering layer*). Dies bezeichnet die Sensoren, die direkt Kontextinformationen messen können.

Die Trennung in die einzelnen Ebenen erfolgte nach den unterschiedlichen Phasen der Kontextverarbeitung. Hierbei ist besonders beachtenwert, dass die Architektur Kontextmanagement als eigenständige Aufgabe sieht, die zwischen der Kontextgewinnung und der Nutzung steht; diese Aufgabentrennung erscheint sinnvoll und wird in den zu erarbeitenden Architekturansatz übernommen.

¹HENRICKSEN UND INDULSKA verwenden in der Abbildung eine abweichende Semantik für den Begriff der »Situation«, auf den allerdings nicht näher eingegangen werden soll

Key:

- > Synchronous communication
- - -> Asynchronous communication

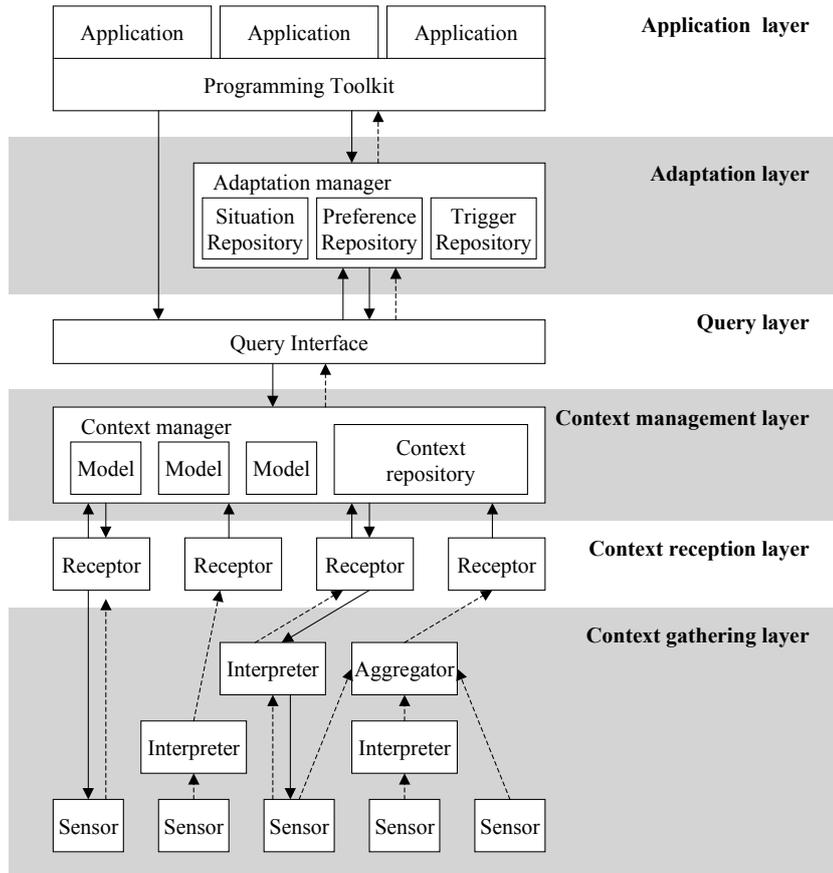


Figure 6. Architecture.

Abbildung 9.1: Architekturmodell für kontextbewusste Anwendungen. [HI06]

ural and easy to use, and our experience of mapping the CML model to a relational database was a painless one. However, we encountered some unexpected results with the preference model in relation to preferences that assigned frequent indifferent scores. This occurred when the preference scopes incompletely covered the set of possible contexts - for example, when users specify preferences only in relation to CSUs but not to their sub-CSUs.

9.1.2 Adaptive E-Learning-Systeme

Im Bereich der adaptiven Hypermediasysteme für E-Learning hat sich inzwischen eine Quasi-Referenzarchitektur herausgebildet, die von KARAMPIPERIS UND SAMPSON [KS05] zusammengefasst wird. Sie teilen ihre Ebene und eine Laufzeitebene (vgl. Abb. 9.2) in Lernressourcen und ihre Metadaten, Lernziele und in den Metadaten verwandelte Benutzermodelle, das Handling von Präferenzscores und ein Regelmittel, das Regeln für die Auswahl von Inhalten dem Domänenmodell enthält. Die Laufzeitebene besteht aus Komponenten zur Beobachtung des Benutzerverhaltens, zur Auswertung von Regeln und zur Anzeige der Inhalte.

gramming toolkit were very positive, however, we do plan to extend the toolkit in the future to provide improved handling of the indifferent and veto scores.

Overall, the case study showed that our conceptual framework and software architecture are extremely successful in terms of our original goal of facilitating the development of context-aware applications that use the development of context-aware applications. The use of the model mean that we are able to easily evolve the underlying context model without changing the source code, and both the developer and the end users can easily adapt and use the model of communication channels simultaneously by adding the preferences. This feature facilitate the types of experimentation that are necessary to gain a better understanding of the most compelling uses of context information and to explore the usability problems de-

2. Adaptive Educational Hypermedia Systems: A Literature Review and Discussion

Current state-of-the-art adaptive educational hypermedia systems such as AHA! (De Bra et. al., 2002), OntoAIMS (Aroyo et. al., 2003), The Personal Reader (Dolog et. al., 2004), WINDS (Kravcik and Specht, 2004), *Referenzarchitektur für situationsbewusste lernunterstützende Systeme* (Kapel, 2004), ACCY (Daggar et. al., 2005) are based on the Adaptive Hypermedia Application Model (AHAM) (De Bra, Houben and Wu, 1999).

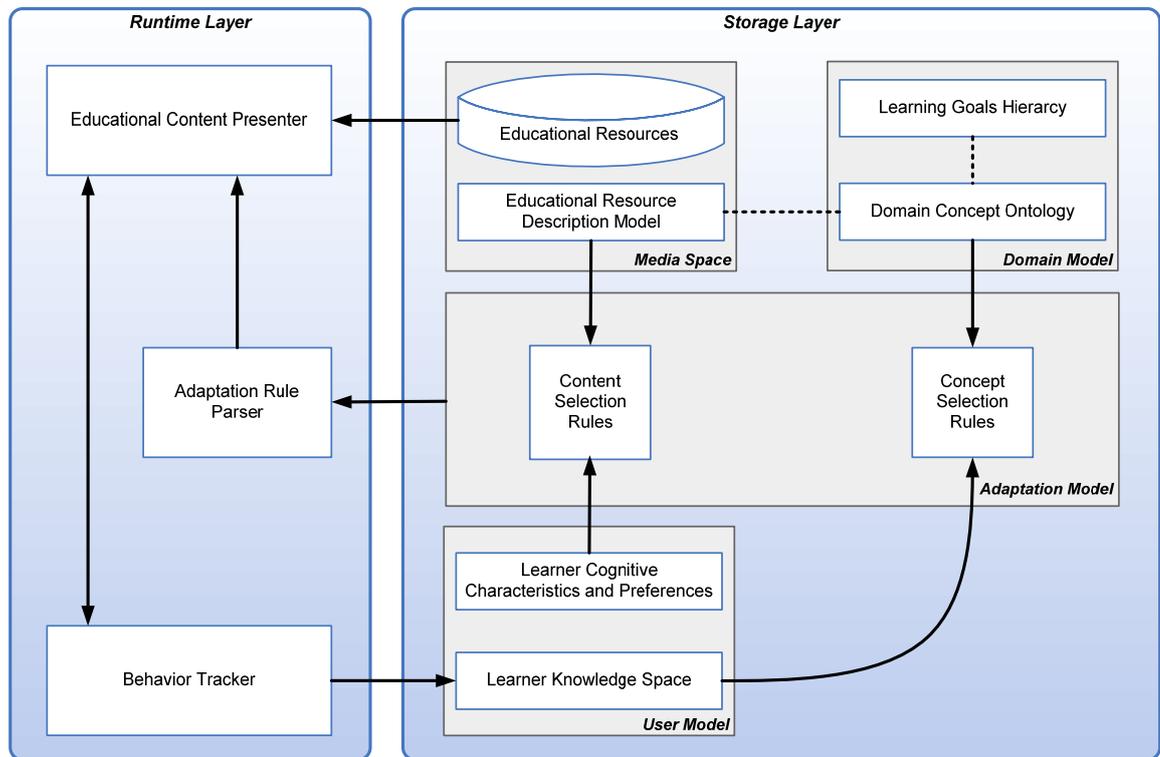


Figure 1: Generalized Architecture of Adaptive Educational Hypermedia Systems

Abbildung 9.2.: Referenzarchitektur für Adaptive Educational Hypermedia Systems [KS05b]

The AHAM builds upon the Dexter model (Halasz and Schwartz, 1994), that is, a common model for hypertext-based systems that was designed for general purpose adaptive web applications. The AHAM model refines the Dexter model so as to be used for educational purposes and extends the hypertext resources to include the full variety of hypermedia objects. The AHAM model consists of two main layers, namely, the *run-time layer* which contains the adaptation engine that performs the actual adaptation and the *storage layer*, which stores information for later use. The design process of an AEHS involves four key steps (Brusilovsky, 2003):

1. *Designing the Domain Model*, that is, the process of designing a hierarchy of learning goals as well as a student hierarchy. The Domain Concept Ontology for describing the subjects in a particular domain is a standardization (z.B. im Rahmen von SCORM) auch Initiativen, die auf (dienstorientierte) Referenzarchitekturen abzielen. Die wichtigsten hierbei sind die *Open Knowledge Initiative* des MIT und die *User Model Schema* Struktur von Carnegie Mellon University.
2. *Designing the User Model*, that is, the process of designing the Learner Knowledge Space, as well as, designing the model for learner's cognitive characteristics and preferences. For the design of the Learner Knowledge Space, there exist two main approaches, the *overlay modeling* (Paiva and Self, 1995) where the learner's state of knowledge is described as a subset of the Domain Concept Ontology and the *stereotype modeling* (Paiva and Self, 1995) where the learner's state of knowledge is described as a set of characteristics of a certain class.
3. *Designing the Media Space*, that is, the process of designing the educational resource description model. This model describes the educational characteristics of the learning resources e.g. the learning resource type, or its difficulty, as well as structural relationships between learning resources e.g. if a resource requires another resource. For each learning resource contained in the Media Space a set of related concepts from the

auf das formelle Lernen konzentriert.

Ein weiterer dienstorientierter Ansatz ist der *Learning Services Stack*, der sich aus dem Architekturprinzip von Kommunikationsprotokollstapeln (wie dem ISO/OSI-Modell oder dem TCP/IP-Stack) ableitet ([BR03], [WBR04]). Dabei werden drei Hauptgruppen identifiziert (vgl. Abb. 9.3). Auf der untersten Ebene ist die Infrastruktur, was eine Standard-Web-Service-Infrastruktur bezeichnet. Darüber sind die Lerndienste, die wiederum in drei aufeinander aufbauende Untergruppen aufgeteilt sind: Basisdienste (wie Persistenz, Authentifizierung, Rechtemanagement, aber auch Workflow-Funktionalität), allgemeine Anwendungen (Wissens- und Content-Management, Benutzerprofilverwaltung und die Sequenzierung von Lernmaterialien) und Werkzeuge (u.a. für die Präsentation von Lerninhalten, für Tests, Tutorenunterstützung und Kollaboration), die für bestimmte Anwendungsbereiche spezialisierte integrierte Schnittstellen bereitstellen will. Die oberste Gruppe bilden die Komponenten mit Benutzerschnittstelle, die aufgeteilt sind in Autorenwerkzeuge, Bereitstellung von Lernmaterialien und die Verwaltung der Lernprozesse.

9.1.4. Enterprise Knowledge Infrastructures

Als Ablösung von monolithischen Wissensmanagementsystemen hat MAIER [MHP05] eine Referenzarchitektur für sog. »Enterprise Knowledge Infrastructures« entwickelt, die die Grundlage für die Unterstützung von Wissensarbeitern bilden soll. Es ist ein Schichtenmodell, das aus folgenden sechs Schichten besteht (vgl. Abb. 9.4):

- **Zugriffsdienste.** Diese Ebene besteht aus Diensten zur Authentisierung von Benutzern und die Aufbereitung für unterschiedliche auf die Infrastruktur aufsetzende Anwendungen oder auch direkt für bestimmte Endgeräte.
- **Personalisierungsdienste.** Auf dieser Ebene sieht MAIER diverse Portallösungen wie personen-, prozess-, projekt- oder rollenorientierte Portale.
- **Wissensdienste.** Diese Ebene besteht aus den vier Teilen Entdeckung (Suche, Visualisierung, Navigation), Veröffentlichung (Strukturierung, Kontextualisierung), Kollaboration (Kompetenzmanagement, Community-Unterstützung) sowie Lernen (Entwicklung von Lerninhalten, Kursmanagement, Tutorenlösungen).
- **Integrationsdienste.** Diese Ebene führt die unterschiedlichen darunterliegenden Systeme konzeptuell zusammen, und zwar auf der Ebene der Begrifflichkeiten, der Benutzeridentität, der Funktionen und der Prozesse. Dies entspricht einer Mediatorrolle in der klassischen Informationsintegration.
- **Infrastrukturdienste.** Diese Schicht kapselt die darunterliegenden Dienste technisch, so dass eine einheitliche Schnittstelle für Suche, Speicherung und Überwachung existiert. Dies entspricht der Ebene der Wrapper.

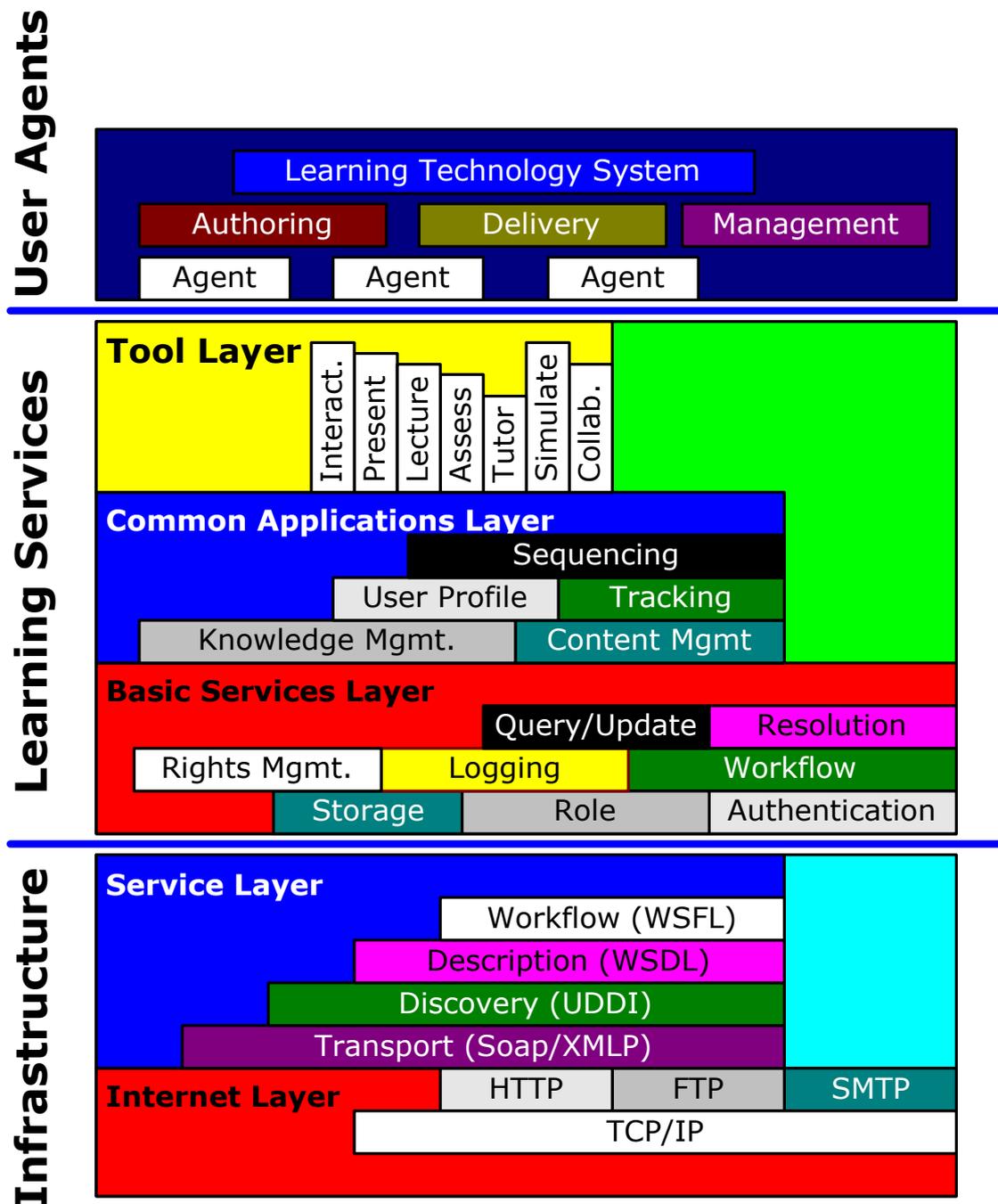


Abbildung 9.3.: E-Learning Framework [WBR04]

© Copyright 2001, Carnegie Mellon University

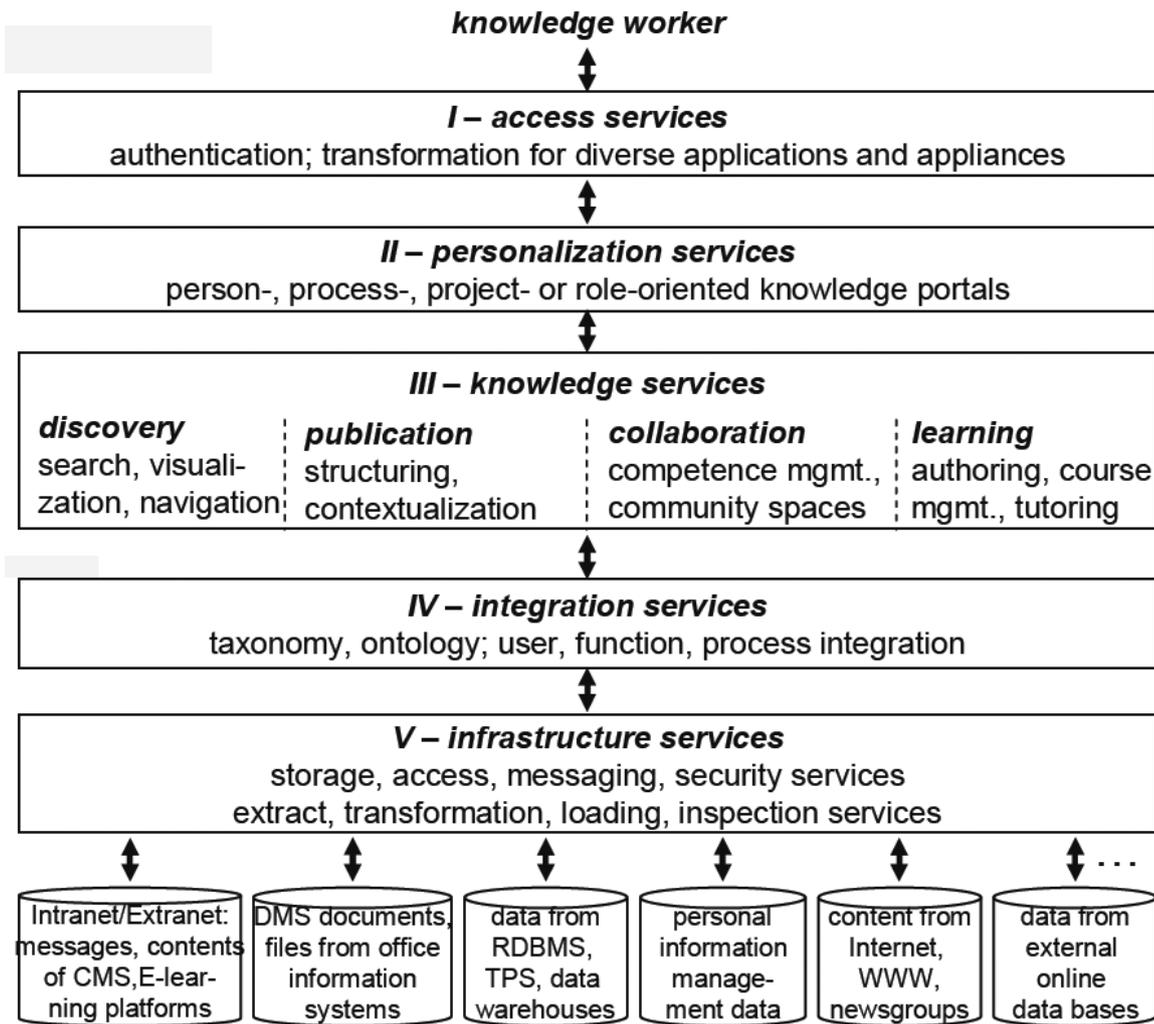


Abbildung 9.4.: Enterprise Knowledge Infrastructure [MHP05]

- **Daten- und Wissensquellen.** Hier finden sich existierende Daten- und Wissensquellen im Unternehmen wieder, wie z.B. Intranet, Dokumentenmanagementsysteme, Datenbank- und Data-Warehouse-Systeme oder auch externe Internetquellen.

Dieses Modell bietet eine klare Trennung zwischen den für unterschiedliche Zwecke nutzbaren Wissensdiensten und den darauf aufsetzenden Anwendungen (die im Modell sehr portalfokussiert sind). Diese Trennung werden wir im folgenden weiterverfolgen. Allerdings ist dabei zu beachten, dass wir i.d.R. von existierenden Portalen auszugehen haben (wie z.B. Lernmanagementsysteme), die wir nur »minimalinvasiv« anpassen wollen,

so dass sie situationsbewusste Dienste nutzen.

9.2. Grundstruktur der Referenzarchitektur

Wie in Abschnitt 7.4 ausgeführt, bildet die Grundlage für die Referenzarchitektur ein ontologiezentrierter Architekturstil, der von einer Menge von Diensten ausgeht, die konzeptuell über eine gemeinsame und »ausführbare« Domänenontologie gekoppelt sind. Für einen solchen Architekturstil im Kontext eines Domain-Engineering-Ansatzes eignet sich eine Ableitung der Architektur aus Anwendungsfällen wenig; hier ist eine Ableitung aus dem Domänenmodell sinnvoller [TA01].

Weiterhin bietet sich als Ordnungsprinzip für die Anordnung der Dienste in der Architektur eine Schichtenarchitektur als Grundmodell an, bei dem die Dienstkommunikation beschränkt ist, so dass ein Dienst nur Dienste auf derselben oder darunterliegenden Schichten nutzen kann. Diese Beschränkung der »Nutzung« darf angesichts des erforderlichen proaktiven Systemverhaltens (was technisch durch asynchrone Benachrichtigungen umgesetzt wird) nicht in eine Beschränkung der »Aufrufe« übersetzt werden, da ein benachrichtigender Dienst auf einer unteren Ebene durchaus einen Dienst auf einer höheren Ebene aufrufen können muss (in Form von sog. »Callbacks«). Dabei geht die Nutzung weiterhin vom Dienst der höheren Ebene aus, da er die asynchrone Benachrichtigung anfordert.

9.2.1. Infrastruktur

Die Hauptfrage, die nun zu klären ist, ist, welche Arten von Diensten benötigt werden und wie diese sinnvoll in Schichten anzuordnen sind. Eine sinnvolle Orientierung bietet das *Enterprise-Knowledge-Infrastructure-Modell*, das mit der Ebene der »Integrationsdienste« eine Schnittstelle bereitstellt, auf der auf Daten und Informationen mit einer einheitlichen Semantik zugegriffen werden kann.

Für uns bedeutet dies, dass eine solche Schicht, die »unsere Domänenontologie spricht«, die Grundlage für lernunterstützende Dienste bilden muss. Anders als MAIER gehen wir entsprechend dem ontologiezentrierten Architekturstil davon aus, dass die wesentlichen Datenquellen bereits semantisch homogen sind (also der Domänenontologie entsprechen); dies teilt sich auf (vgl. Abb. 9.5) in die Instanzen der Informationsontologie (Informationsartefaktverwaltung), der Kontextontologie (Kontextverwaltung), der Portfolioverwaltung und den übrigen Teilen der Domänenontologie (die von einem Ontologieverwaltungsdienst bereitgestellt wird). Insbesondere die ersten beiden speisen sich in der Regel zumindest teilweise aus **externen Quellen**, die sicherlich nicht auf unsere Domänenontologie abgestimmt sind, so dass hier eine Integrationsschicht sinnvoll erscheint (die für den

Einzelfall offen läßt, ob wir eine virtuelle Integration oder eine materialisierte Integration anstreben).

Damit erscheint sinnvoll, gegenüber dem Modell von MAIER folgende Modifikationen vorzunehmen:

- Oberhalb der Schicht der »Integrationsdienste« wird eine Schicht unter der Bezeichnung **Infrastruktur und Datenhaltung** eingeführt, die eine semantisch integrierte Sicht bereitstellt, aber darüber hinaus auch Persistenzfunktionen wahrnimmt (Speicherung von Artefaktmetadaten und Kontextinformationen). Auf diese Ebene gehören auch noch Dienste, die für Fragen der Authentifizierung und Autorisierung zuständig sind (wie in den E-Learning-Architekturrahmenwerken), um die wir uns allerdings im folgenden nicht weiter kümmern.
- Die Schicht der Integrationsdienste dient nur noch der Anbindung externer Quellen, die nicht von unserem Lernunterstützungssysteme verwaltet werden, behält allerdings ansonsten ihre Funktionen unter der Bezeichnung **Integration**. Dies entspricht im übrigen der Kontextaufnahmeschicht von HENRICKSEN UND INDULSKA.
- Die »Infrastrukturdienste-Schicht« von MAIER wird aufgelöst; ihre Funktionen werden bereits von der **Infrastruktur und Datenhaltungsschicht** wahrgenommen.
- Es bleiben die externen Quellen (im Modell von MAIER nicht separat bezeichnet), die einer Verallgemeinerung der Kontextmessschicht von HENRICKSEN UND INDULSKA entsprechen.

9.2.2. Situationsbewusste Dienste

Auf diese Infrastrukturschicht setzen nun Dienste auf, die MAIER als Wissensdienste bezeichnet. Hier lohnt allerdings eine weitere Differenzierung. Betrachten wir die funktionalen Anforderungen aus Abschnitt 7.1, so brauchen wir Dienste, die zusätzliche, kontextabhängige Relationen (und teilweise Entitäten) in der Domänenontologie berechnen (vgl. Abb. 9.6):

- *Kompetenzlückenberechnung*. Für einen Benutzer muss aus seinem Kontext und den Anforderungsprofilen berechnet werden, welche Kompetenzen ihm im aktuellen Kontext fehlen (Relation »needs-competency«, Anforderung AF1).
- *Auswahl von Lerngelegenheiten*. Aufbauend auf dieser Relation müssen Lerngelegenheiten bestimmt werden, die situativ einen Kompetenzerwerb ermöglichen (hierzu eine Relation »is-relevant-for«, Anforderung AF2). Bei der Empfehlung von Kollegen hat eine Interessensabwägung zu erfolgen.

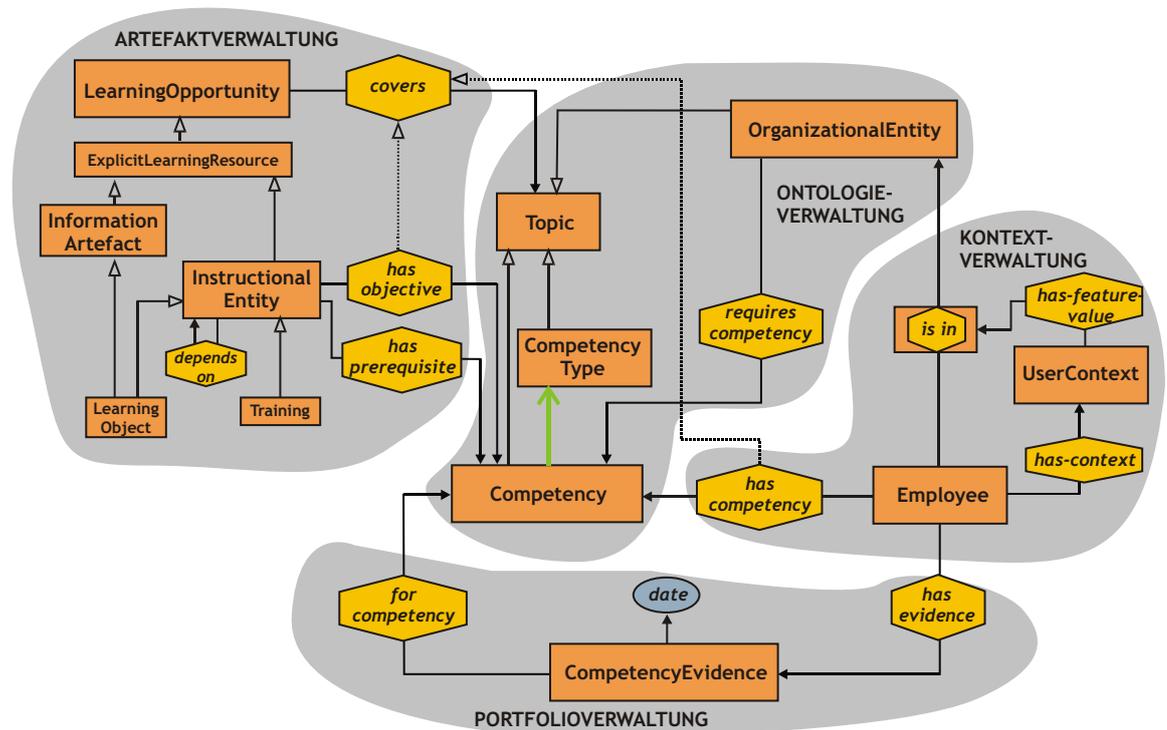


Abbildung 9.5.: Infrastrukturdienstidentifikation anhand der Domänenontologie in Abb. 8.10

- *Navigationsempfehlungen.* Für einzelne Lerngelegenheiten sind empfohlene Navigationsmöglichkeiten zu generieren (Relation »is-suggested-for« zwischen Lerngelegenheiten (Anforderung AF7)).
- *Lernprogrammzusammenstellung.* Eine einfache Zuordnung von Lerngelegenheit zu Benutzern ist oft nicht möglich. Hier ist eine zusätzliche Entität Lernprogramm erforderlich, die eine geordnete Liste von Lerngelegenheiten repräsentieren und die zu berechnen ist (Konzept *LearningProgram* und zugehörige Kompositionsrelation *consists-of-learning-object* zu den Lernobjekten, Anforderung AF4).

Auf der anderen benötigen wir Dienstfunktionalität, die in die Arbeitsumgebung des Lernenden eingreift:

- *Empfehlungspräsentation.* Das System muss proaktiv und auf die Situation passend die berechneten relevanten Lerngelegenheiten anzeigen (Anforderung AF3).
- *Kommunikationsvermittlung.* Um die Kommunikation mit Kollegen zu beeinflussen,

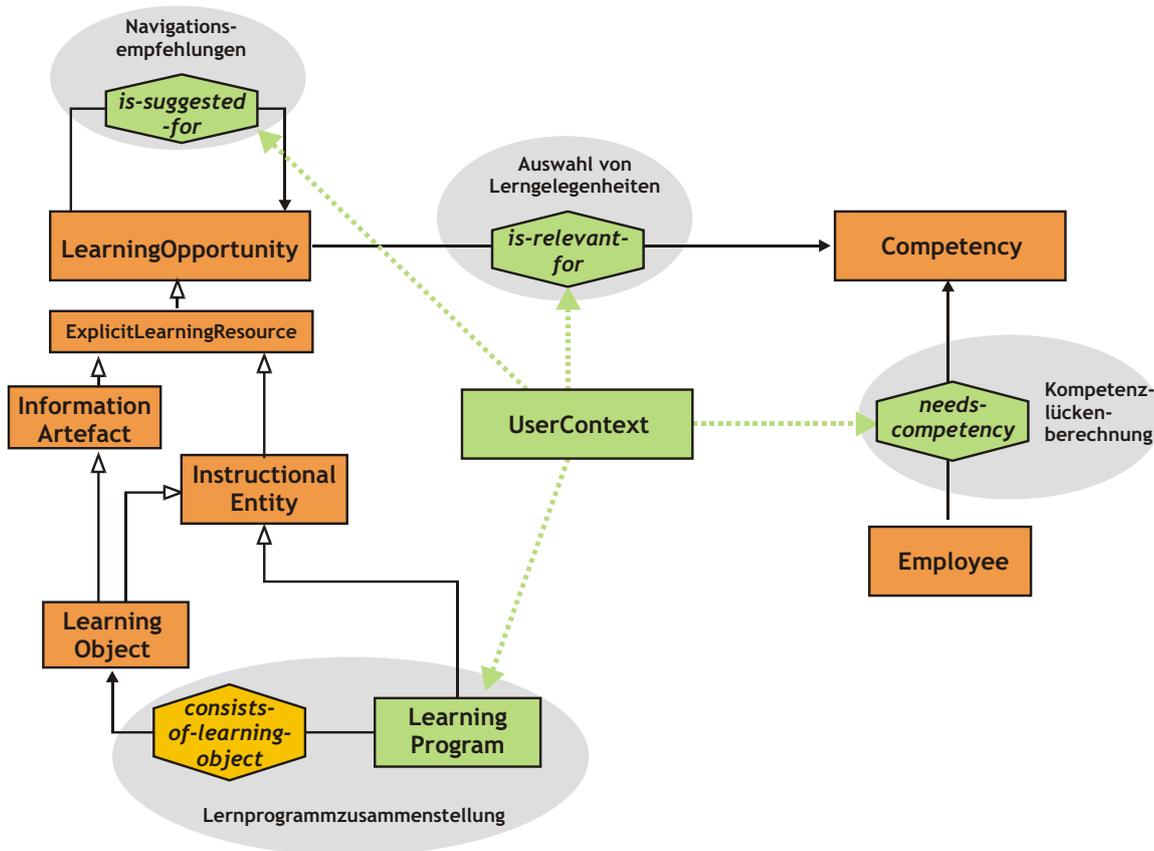


Abbildung 9.6.: Situationsbewusste Dienste zur Berechnung von kontextabhängigen Relationen und Entitäten in Ergänzung zu Abb. 8.10

muss in die entsprechenden Kommunikationsanwendungen eingegriffen werden (Anforderung AF5, AF6).

Wie man erkennen kann, bauen die proaktiven Funktionen auf die rein »berechnenden« Funktionen auf, so dass es sinnvoll ist, zwei Dienstsichten zu wählen: eine Schicht mit **Situationsbewussten Diensten**, die das Domänenmodell anreichern um situationsabhängige berechnete Relationen und eine darauf aufbauende Schicht von Diensten zur »Lernkoordination«, die über Art und Zeitpunkt des Eingriffs in die Umgebung des Benutzers entscheiden und damit pädagogische Strategien implementieren. Beide Schichten sollen – ebenso wie die Infrastrukturschicht – von allen darüber liegenden Schichten zugreifbar sein, da die situationsbewussten Dienste und die proaktive Funktionalität nicht von allen benötigt wird.

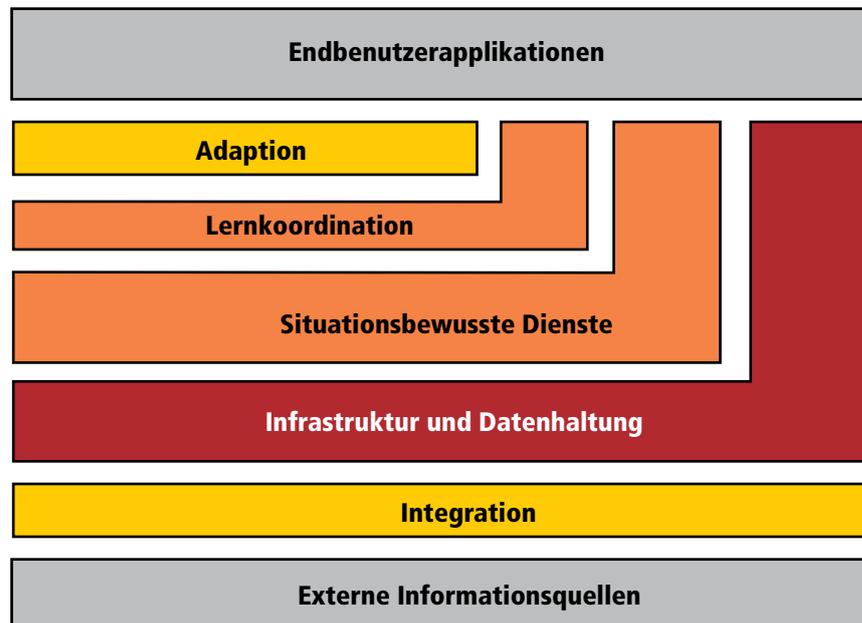


Abbildung 9.7.: Referenzarchitektur lernunterstützender Systeme

Weiterhin werden Werkzeuge benötigt, die mit dem Benutzer interagieren können (Schicht der **Endbenutzerapplikationen**). Da wir aber nicht alle existierenden Endbenutzeranwendungen ersetzen wollen (siehe auch Anforderung AF6), sondern sie nur gezielt anpassen wollen, ist eine zusätzliche dazwischen geschobene Schicht sinnvoll, die sich um die **Adaption** kümmert. Dies entspricht auch der Philosophie der betrachteten Architektur von adaptiven Hypermediasystemen.

Diese Grundstruktur soll im folgenden Abschnitt durch die Definition von Diensten noch etwas weiter konkretisiert werden.

9.3. Dienste der Referenzarchitektur

Nachdem mit Abb. 9.7 nun ein grobes Architekturmodell existiert (in Abb. 9.8 nochmals mit Aufgaben der jeweiligen Schichten und ihren Dekompositionsprinzipien abgebildet), können darauf aufbauend die Dienste und ihre Verantwortlichkeiten konkretisiert werden. Hierzu wird informell die Funktion, die vom Dienst zu erbringen ist, umrissen, bevor eine abstrakte Schnittstelle gegeben wird, die die essentiellen Funktionen semi-formal beschreibt. Diese Schnittstellennotation orientiert sich an der Syntax von Java 1.5 unter Verwendung der dort eingeführten generischen Datentypen; so bezeichnet z.B. `List<URI>`

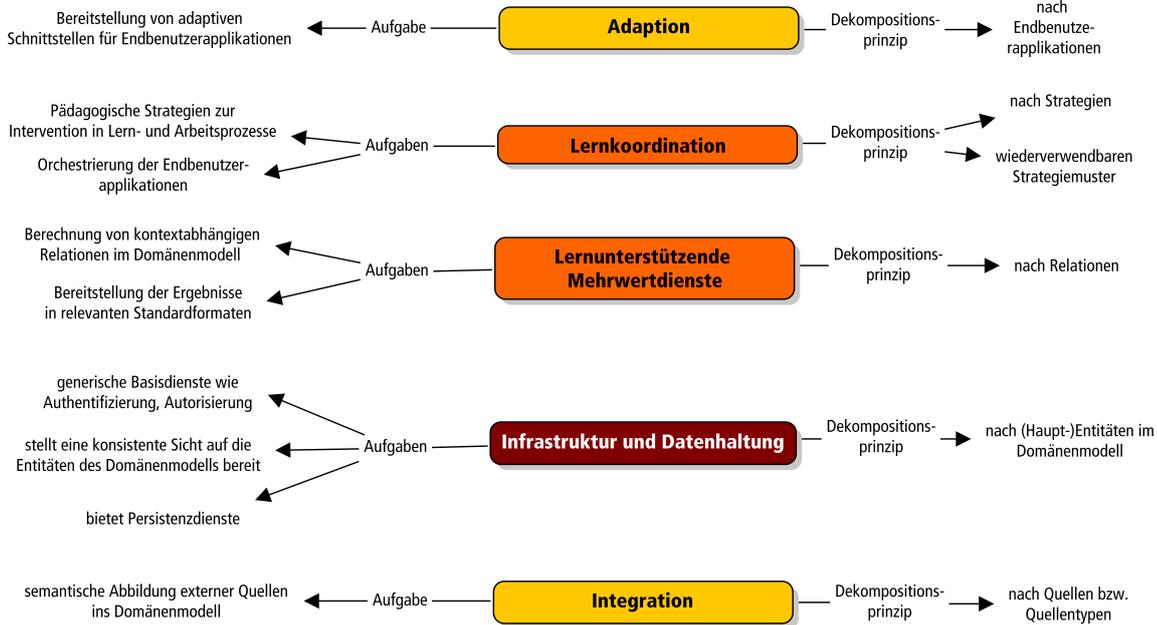


Abbildung 9.8.: Aufgaben der Schichten der Architektur und anzuwendenden Dekompositionsprinzipien

eine Liste von Objekten vom Typ `URI`.

9.3.1. Infrastruktur und Datenhaltung

Die Aufgabenteilung auf der Infrastrukturebene orientiert sich an einer Aufteilung der Instanzen der Domänenontologie aus Abb. 9.5.

Artefaktverwaltung: Lernobjekt- und Dokumentenrepositorien

Die Artefaktverwaltung ist für die Bereitstellung der Informationsartefakte zuständig. Dies können sowohl Lernobjekte sein, die üblicherweise von Learning-Content-Management-Systemen (LCMS) verwaltet werden, als auch Gebrauchsdokumente, wie sie üblicherweise von Desktop- oder Unternehmenssuchmaschinen bereitgestellt werden. Hier ist anzustreben, dass zum einen ein einheitlicher Zugriff auf die unterschiedlichen Arten von Artefakten ermöglicht wird und zum anderen möglichst auf existierende Repositorien zurückgegriffen werden kann. Dies wird es im allgemeinen erforderlich machen, dass die Artefaktverwaltung zusätzliche Metadaten speichern muss (z.B. für Lernobjekte die in Kompetenzen ausgedrückten Lernziele und -voraussetzungen).

Da die Referenzarchitektur auf eine reine Nutzungssicht ausgerichtet ist und den Inhaltserstellungsaspekt ausklammert, wird von den restlichen Diensten nur eine Nur-Lese-Schnittstelle mit einer strukturierten Anfragesprache benötigt. Allerdings sollten zumindest für die vom Dienst selbst verwalteten Metadaten eine Aktualisierungsmöglichkeit vorhanden sein.

Artefaktverwaltung

```
List<URI> getArtefacts(ArtifactQuery)
ArtifactMetadata getMetadata(URI)
void setMetadata(URI, ArtifactMetadata)
Artifact getArtifact(URI)
```

Portfoliodienst

Der Portfoliodienst hat die Aufgabe, die Lernaktivitäten des Lernenden und seine Erfolge/Misserfolge mitzuprotokollieren. Weiterhin ermöglicht er dem Lernenden die Annotation von Lernressourcen und das Anlegen eigener lernbezogener Inhalte. Er stellt eine wichtige Vervollständigung der Architektur dar, da er den aktiven Umgang des Lernenden mit Lernmaterialien und Reflexion fördert. Allerdings wird er, ähnlich wie die Benutzerschnittstellenelemente, die zu seiner Nutzung notwendig sind, im folgenden nicht weiter thematisiert, da zum einen der Bezug zur situationsbewussten Lernunterstützung nur in der Annotation durch Kontextinformationen liegt und zum anderen im Bereich der ePortfolio-Aktivitäten intensiv an einer Standardisierung von ePortfolio-Diensten gearbeitet wird.

Ein solcher Dienst sollte über die Möglichkeit zum Protokollieren von Lernaktivitäten, zur Annotation von Lernressourcen sowie zum Hinzufügen und Verändern eigener Artefakte durch den Lernenden verfügen. Dies schließt selbstverständlich auch Anfragemöglichkeiten mit ein, so dass man zu folgender Dienstschnittstelle kommt.

Portfoliodienst

```

void addActivity(User,UserContext,LearningOpportunity,Date,AssessmentResult)
void addAnnotation(User,UserContext, LearningOpportunity, Annotation)
List<Annotation> getAnnotations(User,LearningOpportunity)
URI createArtefact(User, ArtefactMetadata, Artefact)
void updateArtefact(URI, ArtefactMetadata, Artefact)
void deleteArtefact(URI)
List<URI> getArtefacts(ArtefactQuery)
Artefact getArtefact(URI)

```

Benutzerkontextmanagement

Die Benutzerkontextverwaltung kümmert sich um die Verwaltung der über den Benutzer gesammelten Kontextinformationen. Sie hat die Aufgabe, von den kontextbewussten Diensten die Komplexität des Umgangs mit Kontextinformationen zu verbergen (insbesondere die Imperfektion der gesammelten Daten), indem sie eine konsistente Sicht zurückliefert.

- **Abrufen des aktuellen Kontextes.** Der Standard-Fall für eine kontextbewusste Anwendung ist der Abruf eines bestimmten Teiles des aktuellen Kontextes eines Benutzers, um diejenigen Werte zu ermitteln, die in den Adaptionprozess eingehen. Ein typisches Beispiel ist die Ermittlung der technischen Umgebung (Browser, Bildschirmgröße, Bandbreite, Verfügbarkeit von Lautsprechern), um die Anzeige von Lernobjekten anzupassen.
- **Abrufen von anderen Benutzern in einem bestimmten Kontext.** Gerade bei der Berücksichtigung der sozialen Dimension des Lernens ist es wichtig, den Kontakt zu Kollegen oder anderweitigen Kontakten herzustellen, die »in einem ähnlichen Kontext« sich befinden. Das System kann hier beispielsweise Personen empfehlen, die innerhalb der letzten vier Wochen dieselbe Aufgabe bearbeitet haben. Hierbei ist wichtig, dass dies sowohl benutzerübergreifend als auch die Historie umfassend ist.
- **Aktionen durch Kontextänderungen auslösen.** Gerade für das proaktive Systemverhalten ist es wichtig, dass kontextbewusste Dienste eine zeitnahe Benachrichtigung über Änderungen von bestimmten Kontextinformationen erhalten. Ein Beispiel ist hier der Wechsel von Prozessaktivitäten oder Aufgaben, was ein idealer Zeitpunkt für die Initiierung von Lernprozessen darstellt.

Weiterhin müssen für die Anbindung der Kontextquellen Aktualisierungsschnittstellen zur Verfügung stehen.

Benutzerkontextverwaltung
UserContext getContextForUser(ContextQuery) UserContext getUsers(UserQuery) void registerQuery(ContextQuery, Callback) void unregister(Callback) void addContextFact(ContextFact)

Ontologiedienst

Ein grundlegendes Architekturprinzip war die Nutzung einer gemeinsamen Ontologie als verbindendes Element zwischen den Diensten. Da allerdings diese Ontologie nicht statisch ist, sondern sich kontinuierlich weiterentwickelt (z.B. neue Prozesse, neue Kompetenzen) muss auch sichergestellt sein, dass diese über die unterschiedlichen Dienste hinweg stets konsistent ist. Hierzu dient die Einführung eines Ontologiedienstes, der zuständig ist für die persistente Speicherung der Ontologie, die Sicherung der Konsistenz bei Änderungen und den Zugriff auf die Ontologie. Zusätzlich übernimmt er die zur Umsetzung des verwendeten Ontologieformalismus (z.B. OWL-DL) notwendigen Schlussfolgerungen, was üblicherweise durch Einbindung eines sog. »Reasoners« geschieht.

Als Zugriffsschnittstelle für lesende Zugriffe bietet sich eine deskriptive Anfragesprache wie z.B. SPARQL an [PS06]. Für Aktualisierungen existiert mit der KAON-API [MMS03] ein Satz von Aktualisierungsoperationen, mit dessen Hilfe sich auch komplexe Aktualisierungstransaktionen konstruieren lassen.

Ontologiedienst
Table executeQuery(SPARQLQuery) void update(List<UpdateAction>)

9.3.2. Situationsbewusste Dienste

Im Gegensatz zu den endbenutzerorientierten Diensten treten die Dienste auf dieser Ebene normalerweise für den Lernenden nicht in Erscheinung; sie tragen nur dazu bei, dass

die oben erwähnten Dienste sich besser auf die Situation des Lernenden einstellen können. Diese Dienste sind im wesentliche Matching-Dienste, die die Situation des Lernenden und ihre Anforderungen zusammenbringen mit dem Angebot an verfügbaren Ressourcen, seien sie elektronische Inhalte oder andere Personen (vgl. Abschnitt 9.2.2 und Abb. 9.6):

- **Kompetenzlücke ermitteln.** Auf der Basis der Kompetenzanforderungen der aktuellen Situation und der Kompetenzen des jeweiligen Benutzers kann die Menge der fehlenden Kompetenzen ermittelt werden.

Kompetenzlückenanalyse
List<Competency> getCompetencies(User) List<Competency> getCompetencyGap(User) List<Competency> getCompetencyGap(User,List<OrganizationalEntity>)

- **Lernressourcen finden.** Dieser Dienst kapselt die Funktionen, die zum Auffinden von Lerngelegenheiten dienen, die für die jeweilige Situation relevant sind (also Lernobjekte, Dokumente, Ansprechpartnern etc.). Dabei sollen nicht nur Lerngelegenheiten zurückgeliefert werden, sondern auch der Grad der Übereinstimmung mit der Benutzersituation.

Lerngelegenheitselektion
WeightedList<LearningOpportunity> getLearningOpportunities(User,List<Competency>)

- **Lernprogramme zusammenstellen.** Zum Verständnis einzelner Lernobjekte können Voraussetzungen vorliegen, die zunächst vermittelt werden müssen, so dass ein komplettes Lernprogramm situationsbewusst zusammengestellt werden muss.

Lernprogrammzusammenstellung
List<LearningProgram> getLearningPrograms(User,List<Competency>) List<LearningProgram> getLearningPrograms(User,InstructionalEntity)

- **Kommunikation vermitteln.** Ähnlich wie bei Lernobjekten reicht es auch bei Ansprechpartnern i.a. nicht aus, einfach eine Referenz auszugeben. Vielmehr ist hierbei zu berücksichtigen, dass zwischenmenschliches Lernen immer auch einen informellen Lehrer erforderlich macht, dessen Lehraktivität in seinen Arbeitsprozess integriert werden muss.



9.3.3. Lernkoordination

Diese Ebene koordiniert die Lernunterstützung, indem gezielt die entsprechende Adaptionlogik angesteuert wird. Hier zu findende Dienste reagieren auf Kontextänderungen und übersetzen diese ggf. auf Aktionen in den Endbenutzeranwendungen. Abb. 9.9 zeigt einen Beispielablauf. Der Lernkoordinator wird durch den Benutzerkontextverwalter benachrichtigt über Kontextänderungen. Wie ein Lernkoordinator auf solche Benachrichtigungen reagiert, kann in unterschiedlichen Dienstimplementierungen unterschiedlich ausfallen; um hier den Benutzer durch neue Empfehlungen nicht unnötig zu stören, kann der Lernkoordinator hier entscheiden, ab welcher Art von Kontextänderung eine Aktion erfolgt; durch einen Aufruf der Kompetenzlückenberechnung kann auch eine Veränderung der Lücke als Kriterium herangezogen werden. Entscheidet sich der Koordinator für eine Aktion, ermittelt er passende Lerngelegenheiten. Diese werden dem Lernassistenten zur Präsentation dem Benutzer gegenüber übergeben. Der Benutzer entscheidet, ob er seinen Arbeitsprozess unterbrechen will, um eine Lerngelegenheit wahrzunehmen. Wählt er beispielsweise ein Lernobjekt aus, so kann anschließend die Lernprogrammgenerierung aufgerufen werden, um notwendige Voraussetzungen hinzuzufügen.

9.3.4. Adaption und Endbenutzerinteraktion

Auf der Ebene der Endbenutzerinteraktion sollen weitestgehend existierende Anwendung genutzt werden. Hierzu gehören insbesondere eine Lernumgebung, die das »Abspielen« von Lernobjekten und damit verbundene Aktionen erlaubt, und Kommunikationsanwendungen für die Kontaktaufnahme mit anderen Personen. Zusätzlich wird eine Anwendung benötigt, die direkt mit dem Benutzer interagiert, um ihm Empfehlungen zu präsentieren und daraus eine Auswahl zu ermöglichen.

Hierbei ist für diese Arbeit und die Architekturbetrachtungen in diesem Kapitel nicht so sehr im Vordergrund, wie die Endbenutzerinteraktion aussieht, sondern vielmehr, was

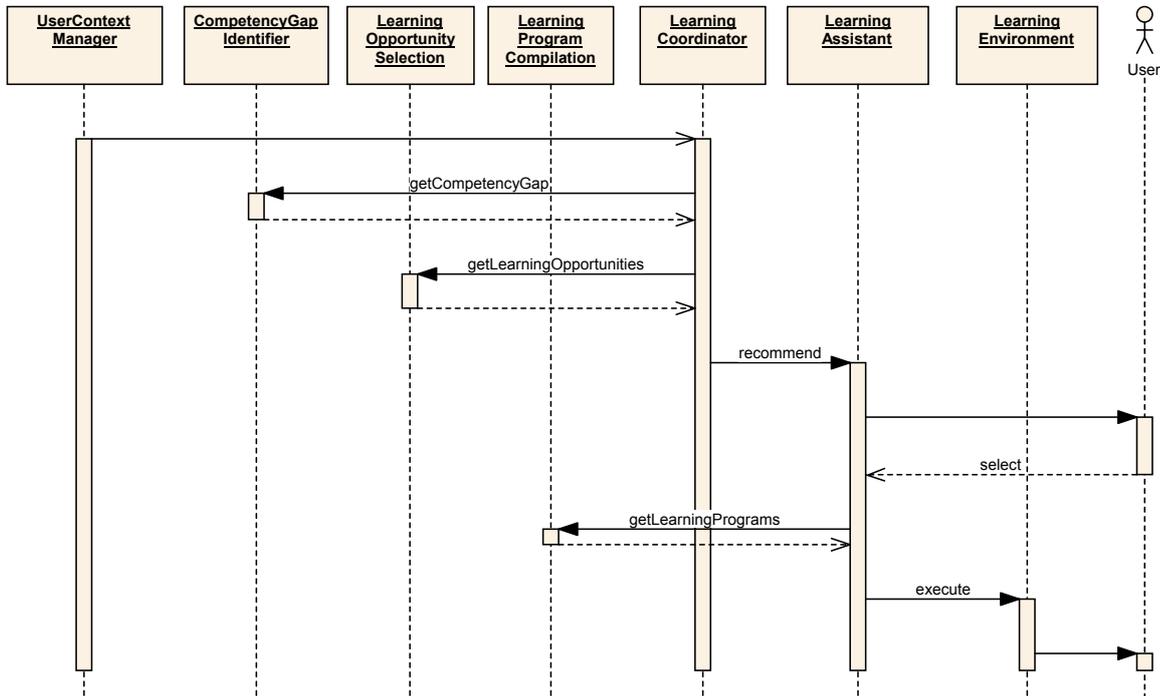


Abbildung 9.9.: Rolle des Lernkoordinators: dynamische Sicht

eine Lernunterstützungsumgebung für Möglichkeiten haben muss, diese Endbenutzeranwendungen anzusprechen und zu steuern. Die im folgenden zu erarbeitenden Schnittstellen sind also Schnittstellen »nach unten« und werden von der Adaptionsschicht bereitgestellt und sind jeweils den dahinter sich verbergenden Anwendungen der Endbenutzerinteraktionsschicht zugeordnet.

Lernumgebung

Dies stellt die primäre Umgebung für den Lernenden dar, innerhalb derer er selbstgesteuerten Zugriff auf verfügbare Lerninhalte hat. Hier besteht auch die Möglichkeit, zu anderen Personen Kontakt aufzunehmen. Dies geschieht zum einen über feste Lerngruppen, zum anderen aber wesentlich über kontextorientierte Auswahl von Kommunikationspartnern und zu Wissensgebieten zugeordneten Tutoren. Existierende (SCORM-kompatible) Lernumgebungen reichen dabei mit ihrer Funktionalität vollkommen aus, allerdings muss gezielt an einzelnen Stellen die Funktionalität kontextbewusst gemacht werden. Primär gehört hierzu die Erweiterung um kontextbewusste Lernobjekte und die automatische Verknüpfung von Lernobjekten mit möglichen Ansprechpartnern.

Lernumgebung

```
void executeLearningObject(LearningObject)
void executeLearningProgramm(LearningProgram)
```

Kommunikationsumgebung

Unter »Kommunikationsumgebung« werden einzelne Dienste zusammengefasst, die es dem Lernenden erlauben, mit anderen Personen in Kontakt zu treten. Hierzu können synchrone (Instant Messaging, Telefon, Konferenzen) wie asynchrone Kommunikationsmedien (EMail, Diskussionsforen) genutzt werden. Auch hier gilt – wie bei der Lernumgebung –, dass das Ziel der Arbeit die Erweiterung dieser Kommunikationsdienste um Kontextbewusstsein ist und nicht der Aufbau einer neuen Infrastruktur. Dieses Kontextbewusstsein setzt im wesentlichen beim Finden der passenden Ansprechpartner an.

Kommunikationsumgebung

```
void openNewMessageTo(List<User>)
```

Lernassistent

Diese Komponente ist zuständig für die proaktive und kontextbewusste Empfehlung von Lernmöglichkeiten in den Arbeitsprozessen. Dabei kann der Lernassistent in eine Fachanwendung eingebettet oder als Einzelanwendung realisiert sein, die periphere Anzeigetechniken nutzen, damit die Benutzer nicht aus ihrer Konzentration herausgerissen werden, wenn sie die Lernmöglichkeit nicht nutzen wollen. Die Art der Präsentation, die vom Lernkoordinator ausgewählt wird, wird als Modalität übertragen.

Lernassistent

```
void recommend(LearningOpportunity,Modality)
```

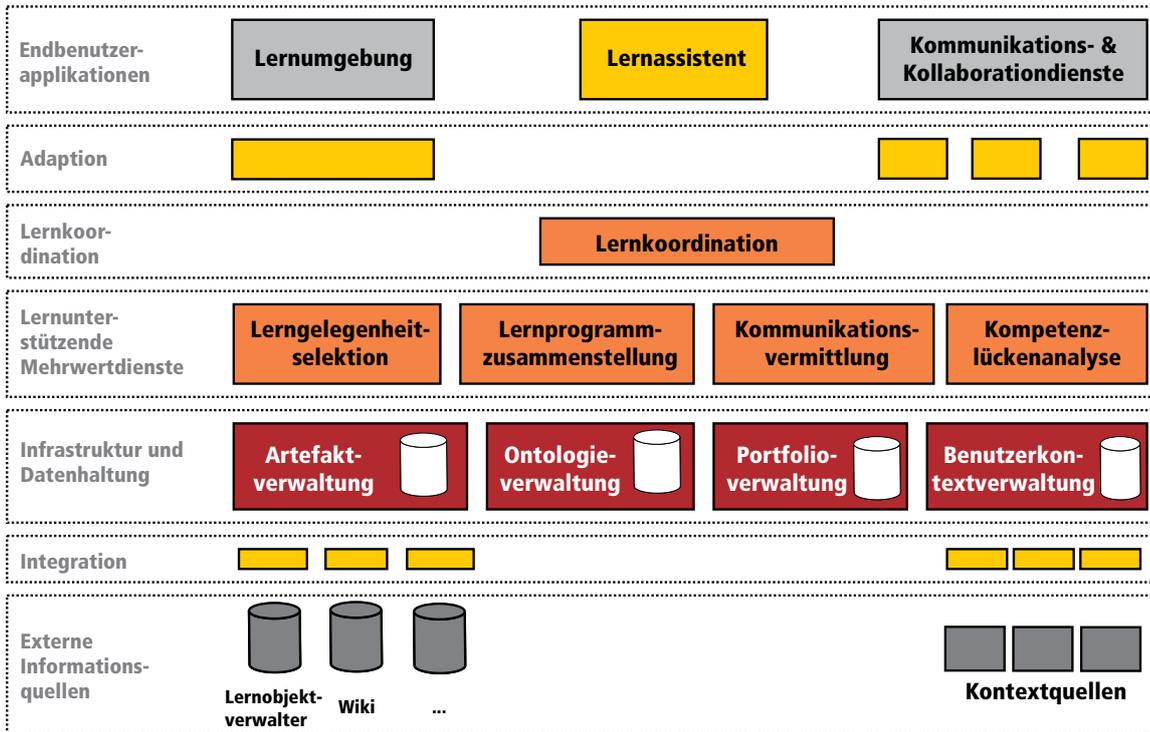
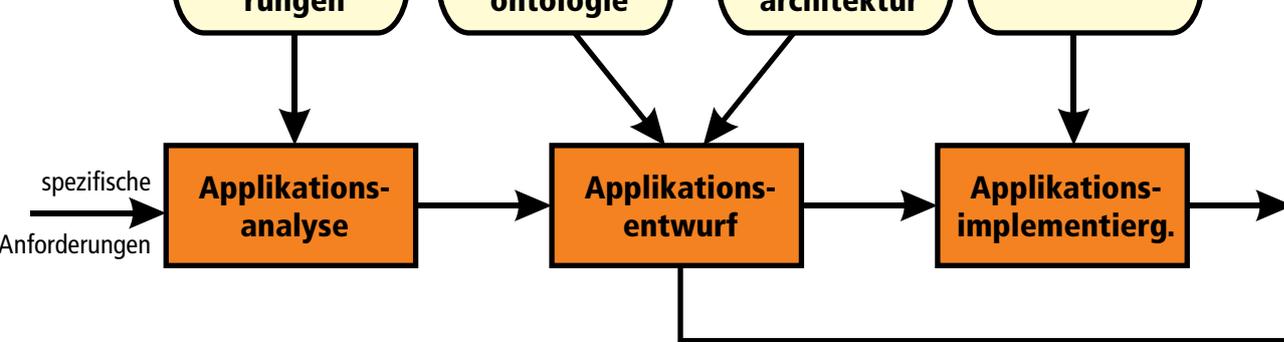


Abbildung 9.10.: Lösungsarchitektur

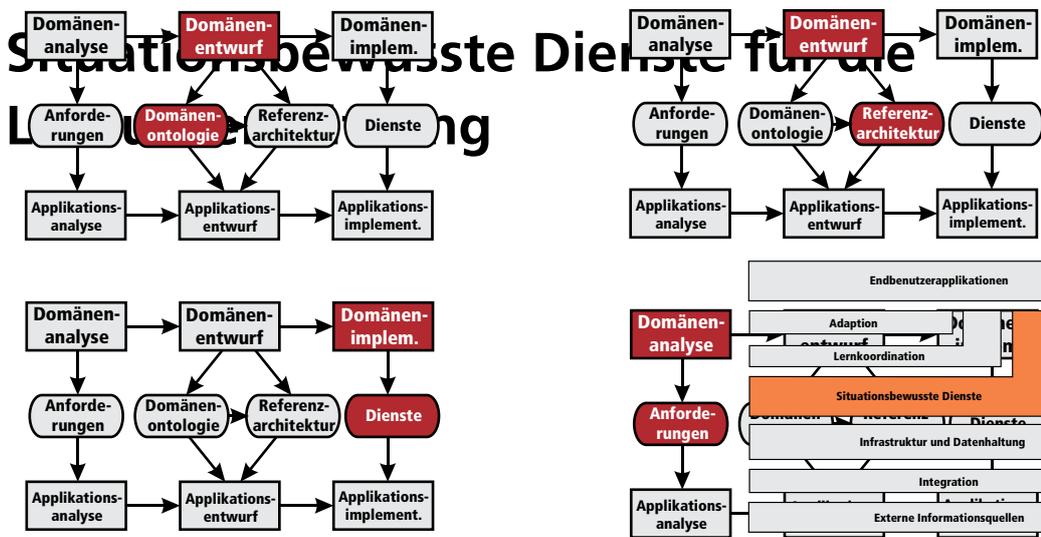
9.4. Fazit: Verfeinerte Architektur

Mit den identifizierten Diensten lässt sich die Architektur wie in Abb. 9.10 darstellen. Auf der linken Seite sind dabei jeweils die Bezeichnungen der Ebenen aus der Referenzarchitektur in Abb. 9.7 zu finden. Die Darstellung ist hier vereinfachend als strikte Schichtenarchitektur dargestellt; es gelten allerdings dieselben Freiheiten wie in der Referenzarchitektur.

Nach der abstrakten Skizzierung der Dienste und ihrer Schnittstellen fehlt jetzt noch eine Implementierung der jeweiligen Dienste (Schritt 3 in der Vorgehensweise des *Domain Engineering*). Für die Infrastrukturschicht erscheint diese bis auf die Kontextverwaltung keine besonderen konzeptuellen Herausforderungen zu bergen, weshalb wir sie im folgenden auch nicht weiter beleuchten. Die Umsetzung der Kontextverwaltung werden wir im vierten Teil der Arbeit intensiver analysieren. In den folgenden beiden Kapiteln widmen wir uns den situationsbewussten Diensten und der Lernkoordination sowie der Anpassungsschicht und dem Lernassistenten.



10.



Nachdem in den vorangegangenen Kapiteln Ontologie und Architektur entwickelt wurden, geht es nun um konkrete Dienste, die für die Lernunterstützung benötigt werden. Dieses Kapitel konzentriert sich hierauf die Schicht der *lernunterstützenden Mehrwertdienste*, die auf die Schicht *Infrastruktur und Datenhaltung* aufbauen, die den Kontext des Benutzers, die verfügbaren Lernangebote und die im Hintergrund das Hintergrundwissen über die Unternehmensumgebung bereitstellt.

Der Fokus liegt hier auf den typischen Problemen, die in den meisten Lernunterstützungssystemen gelöst werden müssen:

- Wie ermittelt man anhand der aktuellen Kompetenzen des Benutzers und der Kompetenzanforderungen, die sich aus seiner Situation ergeben, die zu schließende Kompetenzlücke?
- Wie findet man Lerngelegenheiten, die in der Lage sind, die Kompetenzlücke zu schließen und der Situation des Lernenden angemessen sind? Dies führt auf technischer Ebene auch zu der Frage, wie man Lerngelegenheiten beschreiben muss, so dass sie entsprechend »gefunden« werden können.

- *Wie stellt man im Falle von Lernobjekten diese zu einem Lernprogramm zusammen? Anders als bei klassischen Information-Retrieval-Szenarien reicht es für die Lernunterstützung nicht aus, »relevante« Ressourcen anzuzeigen. Dies ist im wesentlichen darauf zurückzuführen, dass objektive und subjektive Relevanz auseinanderfallen [Swa86]. Während objektiv eine Ressource durchaus geeignet sein kann, eine Wissenslücke zu schließen, ist sie subjektiv nutzlos, weil der Benutzer sie nicht verstehen kann, weil sie inhaltlich Wissen voraussetzt, über das er nicht verfügt. Sie ist dadurch allerdings nicht irrelevant, sondern kann durch vorgeschalteten Wissenserwerb auch nützlich werden. Deshalb ist es sinnvoll, dass das System auch die Aufgabe der sinnvollen Zusammenstellung mehrerer Lernobjekte zu einem Lernprogramm übernimmt.*

Die Lösung dieser Probleme korrespondiert mit der Berechnung der Relationen in der Domänenontologie, wie sie in Abb. 9.6 dargestellt sind: zunächst die Relation *needs-competency* (Ergebnis der Kompetenzlückenberechnung), anschließend die Relation *is-relevant-for* (Auswahl der Lerngelegenheiten) und schließlich *consists-of-learning-object* (Lernprogrammzusammenstellung). Eine Verfeinerung der Domänenontologie ist dabei – wie an früherer Stelle schon angedeutet – unumgänglich, wie z.B. die Identifikation von Kompetenzbeziehungen.

Insgesamt lässt sich der Prozess wie in Abb. 10.1 zusammenfassen: Aus dem Kontext werden die entsprechenden Organisationsentitäten (Prozess, Rolle, etc.) ermittelt, für die Anforderungsprofile (also Mengen von erforderlichen Kompetenzen) hinterlegt sind. Aus diesen erforderlichen Kompetenzen auf der einen Seite und den Kompetenzen des Benutzers auf der anderen Seite wird die Kompetenzlückenberechnung durchgeführt, die zusätzliches Hintergrundwissen mit einbezieht. Daraus ergeben sich die zu entwickelnden Kompetenzen, für die passende Lerngelegenheiten ausgewählt und passend zusammengestellt werden.

10.1. Kompetenzlückenberechnung und Lerngelegenheitsselektion

Der erste Problemkomplex betrifft die Bestimmung dessen, was einem Mitarbeiter für die Bewältigung der aktuellen Arbeitssituation an Kompetenzen fehlt. In einem ersten Schritt soll nun genauer analysiert werden, worin die spezifischen Herausforderungen auf technischer Ebene liegen, bevor anschließend die Domänenontologie entsprechend erweitert und ein Lösungsalgorithmus erarbeitet wird.

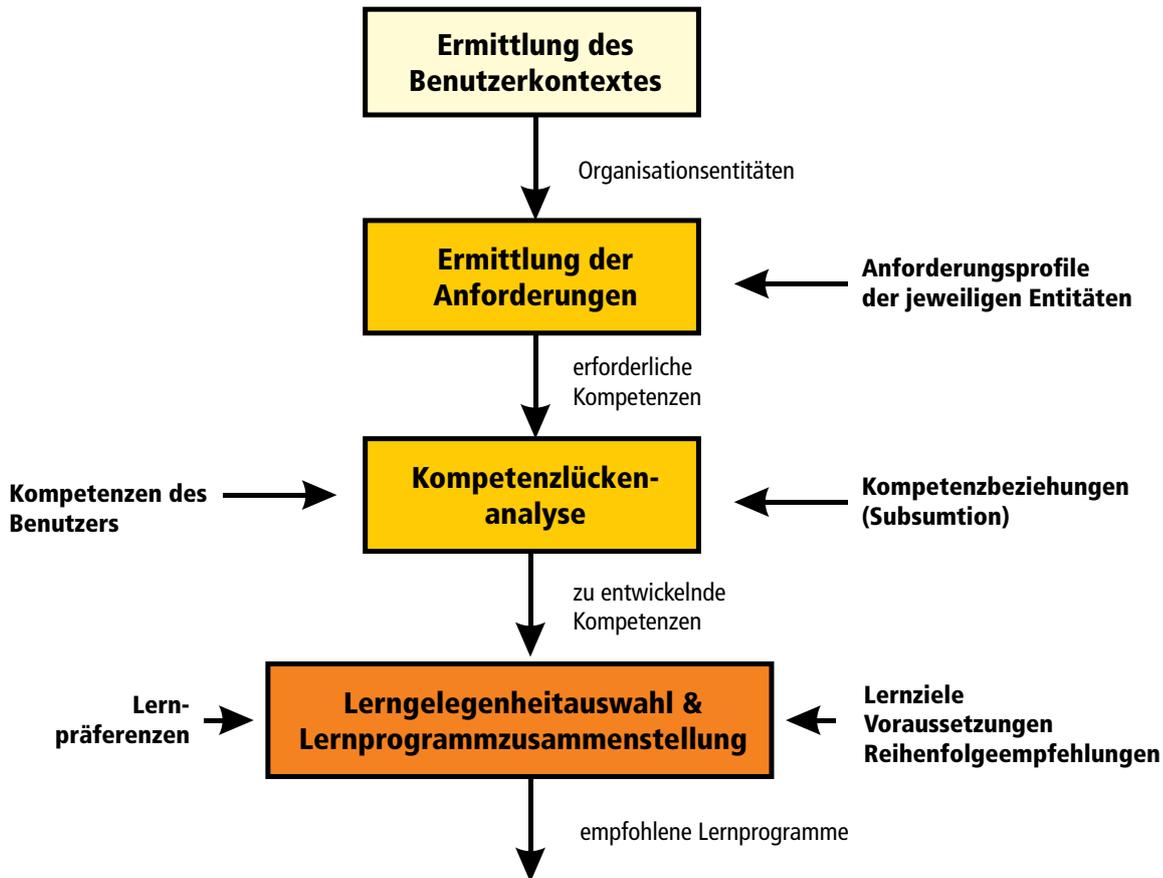


Abbildung 10.1.: Überblick über den Prozess der lernunterstützenden situationsbewussten Dienste

10.1.1. Problemanalyse

Die Berechnung der Kompetenzlücken wie in Abb. 9.6 schematisch dargestellt scheint auf den ersten Blick ein relativ einfacher Vorgang zu sein, bei dem zunächst die erforderlichen Kompetenzen der aktuellen Organisationsentitäten im Kontext des Benutzers berechnet werden und davon anschließend die aktuellen Kompetenzen des Benutzers abgezogen werden (vgl. hierzu die zugrundeliegende Ontologie in Abb. 8.10). Dies könnte beispielsweise mit der Mengendifferenz folgender beider SPARQL-Anfragen [PS06]¹ für einen Be-

¹In der Spezifikation ist die SPARQL-Semantik nur auf reine extensionale RDF-Daten (also ohne Inferenz auf der Basis der Konzepte und Eigenschaften) definiert. Allerdings lässt sich die Semantik von SPARQL unter gewissen Einschränkungen, die hier nicht relevant sind, z.B. auf OWL-DL erweitern (vgl. [HKRS08]); so ist das z.B. im Inferenzsystem KAON2 [Mot06] realisiert. In diesem Sinne wird SPARQL hier und in

nutzer *userURI* geschehen:

```
SELECT    ?competency
WHERE     {{ <userURI> is-in ?entity}
           { ?entity requires-competency ?competency} }
```

```
SELECT    ?competency
WHERE     {<userURI> has-competency ?competency}
```

Wenn man diese Menge der fehlenden Kompetenzen als $\{c_1, c_2, \dots, c_n\}$ bezeichnet, könnte man darauf aufbauend dann relevante Lerngelegenheiten wie folgt selektieren:

```
SELECT    ?l
WHERE     { {?l covers <c1>}
          UNION
          {?l covers <c2>}
          UNION
          ...
          UNION
          {?l covers <cn>} }
```

Will man nur *InstructionalEntities* haben, so läßt sich die Anfrage modifizieren zu:

```
SELECT    ?l
WHERE     { {?l has-objective <c1>}
          UNION
          {?l has-objective <c2>}
          UNION
          ...
          UNION
          {?l has-objective <cn>} }
```

Leider funktioniert das nicht so einfach, und die Hauptursache hierfür ist, dass wir für feingranulare Lernempfehlungen auch von feingranularen Kompetenzkatalogen ausgehen müssen, die unausweichlich mit sich bringen, dass Kompetenzen z.B. auf unterschiedlichen Granularitätsebenen spezifiziert werden und somit keine direkte Übereinstimmung gegeben ist, obwohl das »aus der Natur der Sache« geschlossen werden könnte (in der Kognitionspsychologie ist das als »basic level phenomenon« bekannt [TT91]). Dies deutet darauf hin, dass wir zusätzliches Hintergrundwissen einbringen müssen.

Nehmen wir die Beispielabläufe der Szenarien in Abschnitt 6.2:

- Unsere Projektmanagerin Esther in Szenario 2 hat die Aufgabe bekommen, den Periodenbericht über den Ressourcenverbrauch eines EU-Projektes zu erstellen und bei Abweichungen zum ursprünglichen Plan diesen für die nächste Periode entsprechend zu aktualisieren (Aufgabe »EU Cost Statement«). Daraus ergeben sich die

den folgenden Kapiteln als Anfragesprache auf der *intensionalen* Datenbasis (nach der Anwendung von Inferenzen) verstanden.

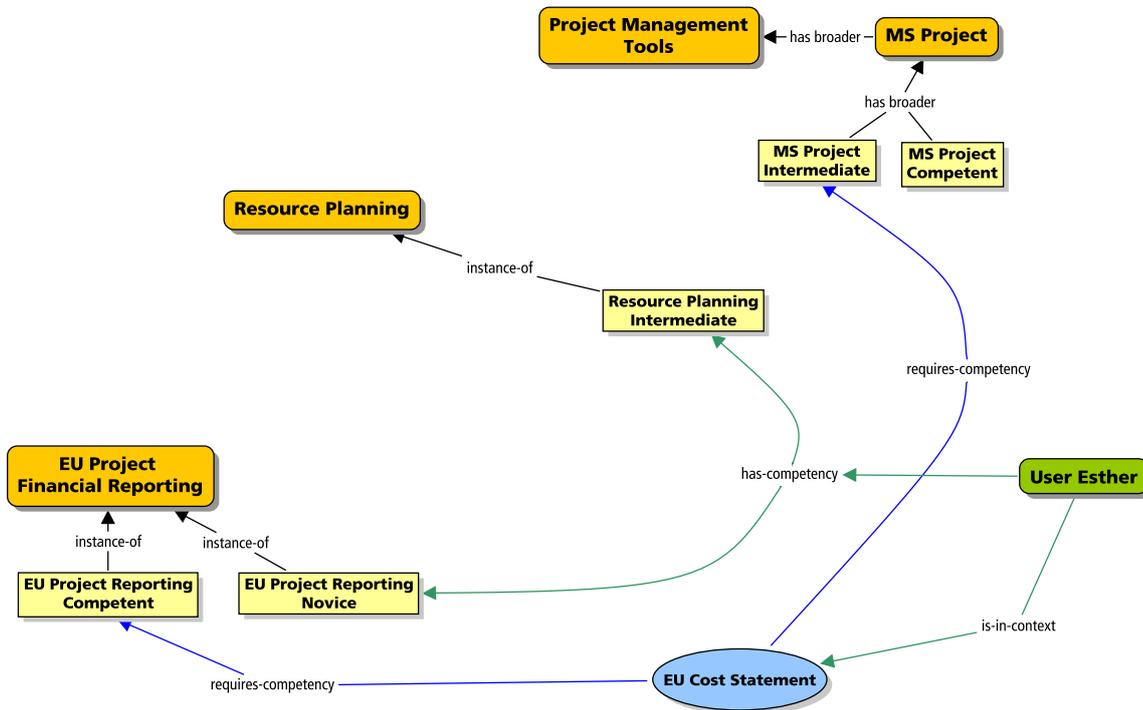


Abbildung 10.2.: Ausschnitt aus einer Kompetenzontologie für das Projektmanagement

Kompetenzanforderungen »EU Project Reporting Competent« und »MS Project Intermediate«. Esther hat bereits ein Seminar besucht (und entsprechende Praxiserfahrung gesammelt), was ihr den Erwerb der Kompetenz »Resource Planning Intermediate« ermöglicht hat (vgl. Abb. 10.2). Ein direkter Vergleich entsprechend einführender Vorgehensweise würde dazu führen, dass sich Esther in *EU Project Reporting* und in *MS Project* weiterqualifizieren müsste, obwohl *Resource Planning Intermediate* im allgemeinen Verständnis immer auch die Bedienungskompetenz von *MS Project* einschließt (obwohl das bislang in der Modellierung nicht sichtbar ist).

- Im Falle des Kundenberaters im Logistikbereich wird man aus Unternehmens- und Personalentwicklersicht sicherlich nicht für jedes Produkt detailliert festlegen wollen, was zu einer kompetenten Beratung zu einem bestimmten Produkttyp alles erforderlich ist (was z.B. Überblickswissen, Preis- und Rabattierungsmodelle, spezielle Bedingungen für spezielle Zielländer etc. umfassen kann). Umgekehrt kann durch den Mitarbeiter durchaus Überblickswissen vorhanden sein und das Wissen um Einschränkungen beim internationalen Versand, jedoch nicht das aktuelle Preismodell präsent sein, so dass für das Lernen bei Bedarf eine feinere Granularität erforder-

lich ist. Dies ließe sich durch zusammengesetzte Kompetenzen erreichen, bei denen aus dem Vorhandensein aller Teilkompetenzen auf die Gesamtkompetenz geschlossen werden kann, so dass man durchaus bei den unterschiedlichen Beteiligten auch unterschiedliche Abstraktions- oder Granularitätsebenen unterstützen kann.

- Für den Entwickler aus unserem dritten Szenario beispielsweise macht es keinen Sinn, ihm für sein neues Projekt eine Einführung in Ontologien zu geben, wenn er bereits intensiv mit der Jena Ontology API gearbeitet hat (Kompetenz »Jena API Programming Expert«), weil er sich diese Kenntnisse informell angeeignet hat.

Diese Beobachtungen legen nahe, dass wir nicht von einer (für uns) »perfekten Welt« ausgehen können, in der Mitarbeiterkompetenzen und Lernziele immer vollständig explizit angegeben werden (also es sind zu jedem Zeitpunkt alle zutreffenden Kompetenzen eingetragen, auch wenn der Kompetenzkatalog verfeinert wird). Diese Annahme ist allerdings bereits in traditionellen Personalentwicklungsansätzen fraglich, wie z.B. [Kun05] zeigt, noch viel mehr in unseren Szenarien zum Lernen bei Bedarf. Deshalb muss jetzt betrachtet werden, wie wir unsere Modellierung durch Hintergrundwissen über den Zusammenhang von Kompetenzen so erweitern, das wir automatisch berechnen zu können, dass eine Kompetenz eine andere subsumiert, sich also »schadlos« durch eine andere ersetzen lässt.

10.1.2. Kompetenzrelationen und Kompetenzsubsumtion

Präziser lässt sich »Kompetenzsubsumtion« wie folgt definieren:

Definition 10.1 (Kompetenzsubsumtion) *Eine Kompetenz X subsumiert eine Kompetenz Y im Zusammenhang einer bestimmten Aussage der Struktur Subjekt Prädikat Y, wenn in dieser Aussage unabhängig vom Subjekt Y durch X ersetzt werden kann, ohne dass der Wahrheitsgehalt in Bezug auf die Realwelt verletzt wird.*

Ein Beispiel für eine bestimmte Aussage ist hierbei *Mitarbeiter A hat eine Kompetenz Y*. X subsumiert Y, wenn für alle Mitarbeiter gilt, dass damit auch die Aussage *Mitarbeiter A hat eine Kompetenz X* richtig ist.

Will man algorithmisch Kompetenzsubsumtion berechnen und ausnutzen, so wird zunächst einmal Hintergrundwissen benötigt, das in Form von Relationen zwischen Kompetenzen bzw. Kompetenztypen ausgedrückt werden kann. Deshalb werden in den folgenden Abschnitten grundlegende Kompetenzrelationen identifiziert, bevor Regeln aufgestellt, ihre Eigenschaften analysiert und erarbeitet werden, wie sich aus ihnen spezifisch für unterschiedliche Zusammenhänge Kompetenzsubsumtion berechnen lässt.

Kompetenzrelationen

Die erste Form von Hintergrundwissen haben wir bereits mit der Ordnung auf Kompetenzniveaus in die Ontologie eingeführt. Hier dürfte unmittelbar einsichtig sein, dass ein Mitarbeiter, der über die Kompetenz C_1 mit Niveau n_1 verfügt auch eine Kompetenz C_2 desselben Kompetenztyps mit niedrigem Kompetenzniveau besitzt: Als Ontologie-Experte verfügt man automatisch auch über die Kompetenz *Ontologie Fortgeschrittener*. Dies gilt analog auch für die Lernziele von Lernobjekten: was vom *Projektmanagement Neuling* zum *Projektmanagement-Experten* führt, führt natürlich auch zu einem Kompetenzniveau *Projektmanagement Fortgeschrittener*.

Wenn man sich reale Kompetenzkataloge anschaut (vgl. z.B. [Bie06]), werden üblicherweise Kompetenzen hierarchisch strukturiert, um größere Kompetenzkataloge noch beherrschbar und wartbar zu erhalten. Auf oberste Ebene kommen dabei oft die Grobeinteilungen Fachkompetenz, Methodenkompetenz und Sozialkompetenz zum Einsatz (vgl. z.B. [Bie06], [ZA05] oder [Kun05]). In [KS06] wurde in Zusammenarbeit mit dem Autor anhand eines Kompetenzkataloges mit Krankenpflegekompetenzen gezeigt, dass monohierarchische Strukturen (also Baumstrukturen) für größere Kompetenzkataloge nicht sinnvoll und vielmehr polyhierarchische Strukturen angemessener sind. Auch wenn in der Praxis oft nicht semantisch sauber modelliert wird, sind dabei die Hierarchiebeziehungen im wesentlichen Generalisierungsbeziehungen gleichzusetzen. Für diese Arbeit dienen **Generalisierungsbeziehungen** im wesentlichen dazu, unterschiedliche Abstraktionsgrade von Kompetenztypen im Kompetenzkatalog zur Verfügung zu haben. Dies ist beispielsweise beim Empfehlen von Ansprechpartnern sehr hilfreich. Typischerweise sucht man Ansprechpartner auf einem höheren Abstraktionsniveau als die Ist-Kompetenzen der Mitarbeiter erfasst werden. Während eine Situation es erforderlich macht, etwas über »Ontologien« zu wissen, sind für die empfohlenen Experten üblicherweise Kompetenzen wie »OWL-Experte« erfasst. Durch die Generalisierungsbeziehung läßt sich dem System der inhaltliche Zusammenhang zwischen den beiden Kompetenzen bekannt machen.

Aus der Vernetzung von Einzelkompetenzen können sich weiterhin auch »höherwertige« Kompetenzen ergeben, was sich am besten durch eine **Kompositionsrelation** ausdrücken läßt. Hierbei ist wichtig, dass diese Komposition meist die Ausprägung der Kompetenzen nicht ignorieren kann. Damit kann die Komposition nicht auf Kompetenztypebene, sondern muss auf Kompetenzebene formuliert werden. Diese Kompositionsbeziehung ist ein wichtiges Instrument, um unvollständig erfasste Kompetenzen zu »vervollständigen« und um den erforderlichen Kompetenzerwerb in Teilbausteine zu zerlegen.

Weitere Relationen zwischen Kompetenzen sind denkbar und können auch in konkreten Situationen sinnvoll erscheinen, so z.B. *Vorgängerrelationen*, wie sie sich beispielsweise zwischen der Bedienkompetenz für unterschiedliche Versionen einer Software ergeben (»Office 2003«, »Office 2007«). Aus Modellierungssicht drücken diese einen sinnvollen Zu-

sammenhang aus; allerdings lassen sie sich für die Kompetenzsubsumtion nur schwer heranziehen, da unklar ist, was beispielsweise aus der Kompetenz »Office 2003 Experte« im Bezug auf den Kompetenztyp »Office 2007« geschlossen werden kann. Dies zu konkretisieren gelingt nur im Einzelfall. Eine andere oft vorzufindende Relation ist die *Voraussetzungsrelation* (*prerequisite relationship*), die ausdrücken soll, dass zum Erwerb einer bestimmten Kompetenz eine andere erforderlich ist. Allerdings findet hier eine Vermischung zwischen Kompetenzen als Beschreibung von Verhaltensdispositionen und pädagogischen Strategien zu ihrem Erwerb statt. Im allgemeinen lassen sich Kompetenzen auf unterschiedlichen Wegen erwerben, auch wenn im Einzelfall sich eine bestimmte Reihenfolge aufdrängt.

Betrachten wir nochmals die Szenarien und die Modellierung der hierzu erforderlichen Kompetenzen, wenn man diese Beziehungen zugrundelegt, so läßt sich das Projektmanagementszenario (vgl. Abb. 10.2) wie in Abb. 10.3 modellieren. Die speziellen Kompetenztypen werden in eine Kompetenztaxonomie (die *has-broader-competency*-Relation modelliert die Generalisierung als Spezialisierung der *has-broader*-Relation zwischen Themen) eingeordnet. Der Zusammenhang zwischen *Ressourcen-Planung* und *MS Project* wird durch die *consists-of*-Relation (Komposition) repräsentiert. Damit läßt sich erkennen, dass Esthers Kompetenz in Ressourcenplanung bereits die *MS Project*-Kompetenz umfasst: Wenn sich eine Kompetenz C aus den Kompetenzen C_1, \dots, C_n zusammensetzt, dann hat ein Mitarbeiter mit Kompetenz C automatisch auch C_1, \dots, C_n und ein Lernobjekt, das C vermittelt, vermittelt auch automatisch C_1, \dots, C_n . Schon beim Umkehrschluss wird es allerdings schwierig, denn damit dieser Umkehrschluss in der gleichen Weise gilt, muss angenommen werden, dass die Komposition auch vollständig angegeben ist. In diesem Fall müßte man davon ausgehen, dass *MS Project competent* und *Gantt-Chart competent* ausreichen, um in Ressourcenplanung fortgeschritten zu sein, was sicherlich nicht zutrifft, da hierzu zusätzliche Kompetenzen im Methodenbereich erforderlich sind. Ein weiteres Problem ergibt sich durch die zeitliche Perspektive: Was passiert, wenn für eine Kompetenz eine weitere Teilkompetenz hinzukommt? Soll man hier auf Versionierung setzen und unterschiedliche Versionen einer Kompetenzen grundsätzlich als unterschiedlich betrachten, solange keine Äquivalenz explizit angegeben ist? Oder grundsätzlich alle als äquivalent annehmen?

Wenn man das Entwickler-Szenario betrachtet (vgl. Abb. 10.4), so ist auch die Generalisierung im Bezug auf die angestrebte Kompetenzsubsumtion differenziert zu betrachten: Für die Kompetenzen eines Mitarbeiters erscheint die übliche Semantik der Generalisierung angemessen; wer über eine speziellere Kompetenz verfügt, der verfügt auch mindestens über die allgemeinere auf demselben Kompetenzniveau (wer kompetent die *Jena API* beherrscht, der ist auch im allgemeinen in *Ontologien* kompetent). Das Umgekehrte gilt üblicherweise nicht, da man die speziellen Eigenheiten nicht kennt. Für Lernobjekte ist dies i.a. nicht anzuwenden: das Speziellere vermittelt i.a. nicht das Allgemeine – und das Allgemeine nicht das Speziellere: ein Kurs über die *Jena-API* will nicht die allgemeinen *Ontologie-Grundlagen* vermitteln, und der allgemeine Kurs geht nicht auf die speziellen

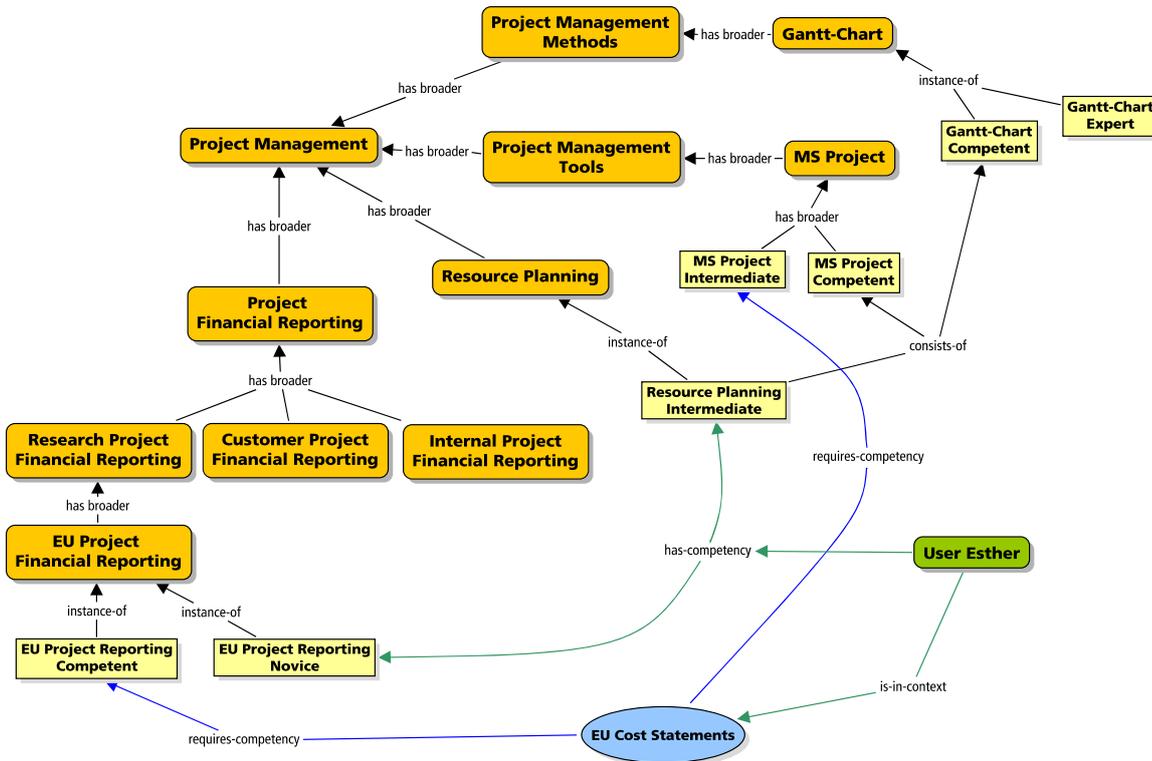


Abbildung 10.3.: Ausschnitt aus einer Kompetenzontologie für das Projektmanagement (erweitert um Kompetenzbeziehungen)

Eigenheiten ein. Damit hat die Generalisierung auch nicht die strengen Eigenschaften einer *is-a*-Beziehung: das Allgemeinere ist nicht immer durch das Speziellere substituierbar.

Diese Analyseergebnisse bestätigen, dass Kompetenzsubsumtion nicht universell, sondern nur abhängig vom Verwendungskontext definiert und formalisiert werden kann, d.h. z.B. im Rahmen der *has-competency* oder *has-objective* Relationen. Voraussetzung hierfür ist eine Formalisierung des Begriff des Kompetenzkataloges.

Formalisierung der Eigenschaften der Kompetenzrelationen und der Kompetenzsubsumtion

Definition 10.2 (Kompetenzkatalog (formal)) Aus dem Domänenmodell ergibt sich, dass eine Kompetenz eine Instanz eines Kompetenztyps und über eine Eigenschaft für das Kompetenzniveau verfügt. Dabei kann eine **Kompetenz** als ein Tupel $k = (t, n)$ repräsentiert werden, wobei $t \in T$ einen Kompetenztyp und n ein Kompetenzniveau repräsentiert. Dieses Kompetenzniveau stammt aus der dem Kompetenztyp zugeordneten Kompetenzskala $s = (N, <)$, die aus einer endlichen

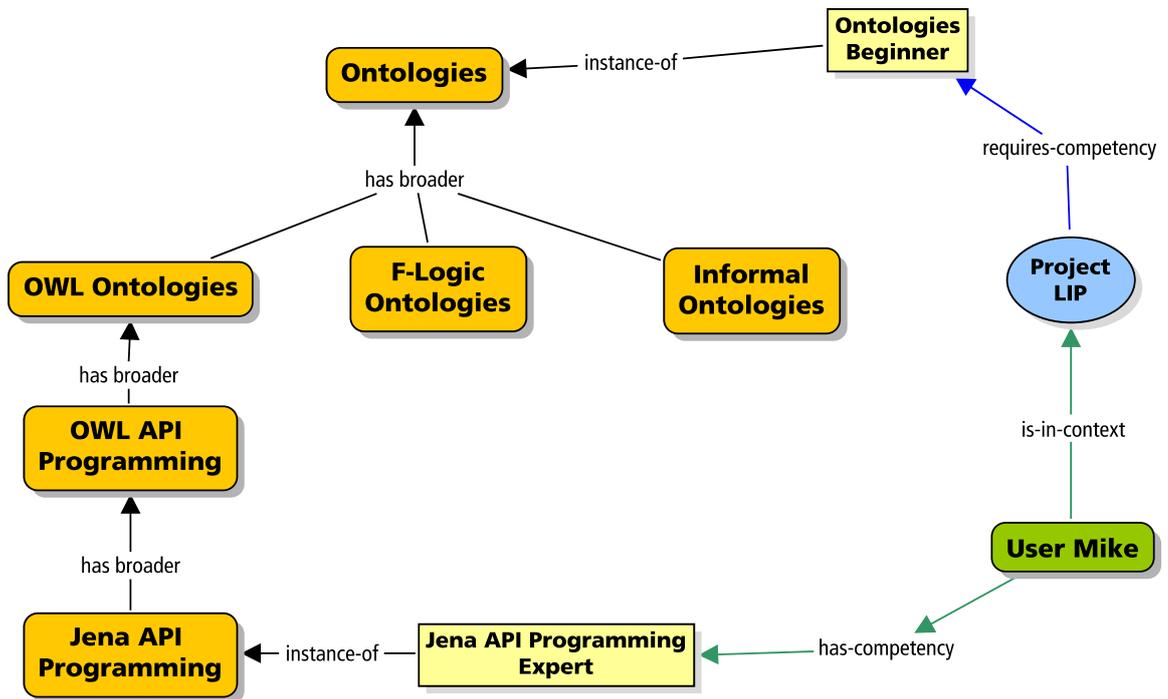


Abbildung 10.4.: Ausschnitt aus einer Kompetenzontologie für einen Software-Entwickler

Menge N von Kompetenzniveaus und einer totalen Ordnung \prec auf dieser Menge besteht.

Damit besteht ein **Kompetenzkatalog** $\mathcal{K} = (T, S, skala, H, C)$, der die Menge von Kompetenzen $competencies(\mathcal{K})$ bilden kann, aus:

- einer Menge T von Kompetenztypen,
- einer Menge S von Kompetenzskalen,
- einer Zuordnungsfunktion $skala : T \rightarrow S$, die jedem Kompetenztyp eine Skala zuweist,
- einer azyklischen Kompetenzgeneralisierungsbeziehung $H \subset T \times T$ (hierbei bedeutet $(t_1, t_2) \in H$, dass t_1 der übergeordnete Kompetenztyp von t_2 ist; H^+ bezeichnet die transitive Hülle von H).
- eine azyklische Kompositionsrelation $C \subset competencies(\mathcal{K}) \times competencies(\mathcal{K})$ (hierbei bedeutet $(k_1, k_2) \in C$, dass k_2 ein Teil von k_1 ist. C^+ bezeichnet die transitive Hülle von C).

Für die Kompositionsbeziehung gilt weiterhin (für die Verträglichkeit mit Kompetenzskalenordnung und der Generalisierung), dass für zwei Kompetenzen $k_1 = (t_1, n_1)$ und $k_2 = (t_2, n_2)$ gilt:

$$(k_1, k_2) \in C \Rightarrow t_1 \neq t_2 \text{ und } (t_2, t_1) \notin H^+$$

was anschaulich bedeutet, dass Kompetenzen nicht aus Kompetenzen desselben Kompetenztyps oder allgemeineren Kompetenzen zusammengesetzt sein dürfen, was eine intuitive Einschränkung darstellt, allerdings bislang nicht in unserer Domänenontologie enthalten war.

Damit haben wir die wesentlichen Grundlagen für die Definition der Inferenzregeln zur Operationalisierung der Kompetenzsubsumtion gelegt. Als erstes ist die *subsumes*-Relation als reflexiv und transitiv zu definieren:

$$(S1) \text{ subsumes}(x, y) \Leftarrow x = y$$

$$(S2) \text{ subsumes}(x, y) \Leftarrow \exists z : \text{subsumes}(x, z), \text{subsumes}(z, y)$$

Skalenordnung

Für die Einbeziehung der Skalenordnung lässt sich folgende Inferenzregeln formulieren.

$$(S3) \text{ subsumes}(k_1, k_2) \Leftarrow t_1 = t_2 \wedge n_1 \prec n_2 \text{ mit } k_i = (t_i, n_i), \text{ d.h. eine Kompetenz desselben Typs mit höherem Kompetenzniveau subsumiert eine Kompetenz mit niedrigerem Niveau (ein Ontologie-Experte besitzt auch die Kompetenz Ontologie-Fortgeschrittener).}$$

Komposition

Für die Einbeziehung der Komposition (vom Ganzen auf die Teile) kann man folgende Eigenschaft ergänzen:

$$(S4) \text{ subsumes}(k_1, k_2) \Leftarrow (k_1, k_2) \in C, \text{ d.h. eine Teilkompetenz wird durch die Gesamtkompetenz subsumiert (z.B. besteht die Kompetenz Ontologie-Experte u.a. aus der Kompetenz »Ontologieeditor beherrschen Fortgeschrittener«, dann wird letztere Kompetenz durch »Ontologie-Experte« subsumiert.}$$

Die Einbeziehung der Komposition (von den Teilen auf das Ganze) kann nicht auf der Ebene der Subsumtionsrelation erfolgen, sondern muss durch zusätzliche Regeln für die Relationen *has-competency* bzw. *has-objective* erfasst werden. Hierbei ist allerdings vorauszusetzen, dass die Teilkompetenzen vollständig aufgeführt sind. Diese Zusatzinformation für die Gesamtkompetenz angegeben werden: *is-fully-decomposed(c)*. Damit lässt sich das wie folgt beschreiben:

(O1) l has-objective $k \Leftarrow \forall k_i \in \{k' \mid (k, k') \in C\} : l$ has-objective k_i und is-fully-decomposed(k).

(C1) u has-competency $k \Leftarrow \forall k_i \in \{k' \mid (k, k') \in C\} : l$ has-competency k_i und is-fully-decomposed(k)

Generalisierung

Für die Einbeziehung der Generalisierung kann man folgende Eigenschaft ergänzen, die nur für Mitarbeiter gilt:

(C2) u has-competency $k = (t, n) \Leftarrow \exists k' \in C : u$ has-competency k' und $(t, t') \in H$

Anwendung der Subsumtion

Für die Anwendung der allgemeinen Subsumtion auf die Relationen *has-competency* und *has-objective* sind zusätzlich noch folgende Inferenzregeln erforderlich:

(OS) l has-objective $k \Leftarrow \exists k' \in C : l$ has-objective k' und subsumes(k, k')

(CS) u has-competency $k \Leftarrow \exists k' \in C : u$ has-competency k' und subsumes(k, k') is-fully-decomposed(k)

10.1.3. Analyse der Kompetenzlücke

Auf der Basis dieser Überlegungen lässt sich nun ein Dienst zur Ermittlung der Kompetenzlücke angeben, also der Menge von Kompetenzen, die für die aktuelle Situation erforderlich sind, aber der Benutzer oder Mitarbeiter nicht besitzt (entspricht der Berechnung der *needs-competency*-Relation aus Abb. 9.6). Die Analyse der Kompetenzlücke erfolgt auf der Basis der Mitarbeiterkompetenzen und der Wissensanforderungen der aktuellen Situation. Hierbei muss auf den Dienst zur Kontextverwaltung zugegriffen werden. Er liefert die aktuellen Kompetenzen und die Zuordnung zu Organisationsentitäten wie Prozessen, Aufgaben und Rollen. Durch einen Aufruf des Ontologiedienstes können die den Organisationsentitäten zugeordneten Kompetenzanforderungen ermittelt werden.

In Pseudo-Code-Notation lässt sich die Dienstfunktionalität wie folgt erbringen:

```
// berechnet die Relation needs-competency(u, c)
Set<Competency> getCompetencyGap(User u)
{
    UserContext c = userContextService.getCurrentUserContext(u);
```

```

Set<Competency> a = ontologyService.getRequirements(c);
Set<Competency> m = getCompetencies(c);
return gap(m, a);
}

```

Die in der Situation erforderlichen Kompetenzen (Pseudo-Methode *getRequirements* des Ontologiedienstes) werden dabei durch eine Anfrage an den Ontologiedienst ermittelt, wie bereits am Anfang des Kapitels als SPARQL-Anfrage skizziert:

```

SELECT    ?competency
WHERE     { {<userURI> is-in ?entity}
           {?entity requires-competency ?competency} }

```

Die eben eingeführte Subsumtion kommt dabei mit der Methode *gap* ins Spiel. Ihre Semantik kann wie folgt beschrieben werden. Sei zunächst Sei $M = \{c \mid m \text{ has-competency } c\}$ die Menge der direkt angegebenen Kompetenzen des betroffenen Mitarbeiters m und A die Menge der Kompetenzanforderungen. Dann ist die Kompetenzlücke:

$$gap(M, A) := A \ominus M := A \setminus M^*$$

wobei M^* das Ergebnis der vollständigen Anwendung der Inferenzregeln (S1) bis (S4), (C1), (C2) und (CS).

Die Berechnung der Funktion kann durch eine Kombination aus regelbasierter und prozeduraler Auswertung erfolgen. Die Subsumtionsregeln (S1) bis (S3) sowie (CS) und (CS2) lassen sich wie folgt umsetzen (als Hornregeln mittels der Jena-Regelsyntax):

[S1: subsumes(?x, ?y) :- ?x = ?y]

[S2: subsumes(?x, ?y) :- subsumes(?x, ?z), subsumes(?z, ?y)]

[S3: subsumes(?x, ?y) :- competency(?x), competency(?y),
has-level(?x, ?levelx),
has-level(?y, ?levely),
is-higher-than(?levelx, ?levely)]

[S4: subsumes(?x, ?y) :- competency(?x), competency(?y),
is-composed-of(?y, ?x)]

[CS: has-competency(?m, ?c) :- has-competency(?m, ?d),
subsumes(?d, ?c)]

```
[C2: has-competency(?m,?c) :- competency(?c), competency(?d)

                               has-competency-type(?c,?ct),
                               has-competency-type(?d,?dt),
                               has-broader-competency(?dt,?dt) ]
```

Diese Regeln werden der Reasoning-Engine als Regeln hinzugefügt, so dass sie automatisch bei Anfragen berücksichtigt werden.

Die Regeln (C1) und (O1) lassen sich wegen des Existenzquantors nicht mit einem Standard-Regelformalismus abdecken, weshalb hierfür auf eine prozedurale Lösung zurückgegriffen wurde:

```
Set<Competency> inferWholeFromParts(Set<Competency> M)
{
  // topologische Sortierung entsprechend der Komposition
  // Annahme: zunächst die Teile
  List<Competency> L = sort(M, is-composed-of);

  while (L is not empty)
  {
    c = L.removeFirst();
    Set<Competency> m = getCompetencies('is-fully-decomposed(c),
                                       is-composed-of(c,?x)');
    if (M.containsAll(m))
    {
      L.append(p);
      M.add(p);
    }
    L.addAll(getCompetencies('is-composed-of(?x,c)'));
  }
  return M;
}
```

Damit lässt sich der Algorithmus zur Berechnung der Kompetenz recht einfach formulieren als:

```
Set<Competency> gap(Set<Competency> M, Set<Competency> A)
{
  Set<Competency> R;
  M = inferWholeFromParts(M);

  for each c in A {
```



Abbildung 10.5.: Ausschnitt aus einer Kompetenzontologie für das Projektmanagement

```

    if (notEmpty(getCompetencies('subsumes(?x,c) AND (?x in M)'))
        R.add(c);
    }
    return R;
}

```

Die Pseudo-Methode *getCompetencies* lässt sich dabei als SPARQL-Anfrage übersetzen:

```

SELECT    ?x
WHERE     { <condition> }

```

Die spezielle Bedingung *?x in M* lässt sich mit $m_i \in M$ umschreiben zu

```

SELECT    ?x
WHERE     { ?x subsumes m1,m2,.. }

```

Nehmen wir das Projektmanagement-Szenario, um zu illustrieren, wie dies am Beispiel aussieht. Aus der Komposition von »Resource Planning Intermediate« ergibt sich die Subsumtionsbeziehung zur Kompetenz »MS Project Competent« (mittels Regel S4). Über Regel (S3) ist dann auch die Kompetenz »MS Project Intermediate« subsumiert, so dass sich als Ergebnis der Inferenz Abb. 10.3 ergibt.

Damit ergibt sich: $gap(\{ Resource\ Planning\ Intermediate, EU\ Project\ Reporting\ Novice \}, \{ MS\ Project\ Intermediate, EU\ Project\ Reporting\ Competent \}) = \{ EU\ Project\ Reporting\ Expert \}$.

10.1.4. Kompetenzbasierte Lerngelegenheitsauswahl

Die andere Seite der Anwendung von Kompetenzen ist die Auswahl von Lerngelegenheiten, die bestimmte Kompetenzen vermitteln können (was der Relation *relevant-for* entspricht⁹. Zunächst soll das für den Fall der Lernobjekte untersucht werden, bevor anschließend andere Artefakte und Personen betrachtet werden.

Auswahl von Lernobjekten

Wie bereits in Abschnitt 8.3 dargelegt, werden Lernobjekte beschrieben durch ihre Lernziele *objectives(LO)* und ihre Voraussetzungen *prerequisites(LO)*; der Wertebereich besteht

jeweils aus einer oder mehreren Kompetenzen. Im Gegensatz zu zahlreichen anderen Ansätzen werden hierbei wohldefinierte und damit maschinenverarbeitbare Entitäten referenziert und keine natürlichsprachlichen unstrukturierten Beschreibungen. Man kann dies zwar durchaus als eine zu starke Vereinfachung sehen, allerdings bleibt zu bedenken:

- Die formale Beschreibung der Lernziele als Einzelkompetenzen muss nicht alle Aspekte einer didaktischen Lernzieles erfassen.
- Die Angemessenheit dieser Beschreibung hängt auch von der Struktur und Modellierungstiefe des verwendeten Kompetenzkataloges ab. Lernziele wie »Teamfähigkeit« sind hier sicherlich vollkommen fehl am Platze.²

Bei der Lerngelegenheitsauswahl können neben den Kompetenzen auch noch andere Faktoren eine Rolle spielen. Insbesondere die Lerncharakteristika (für Lernobjekte) oder die Reifestufe (für Informationsartefakte allgemein), aber auch technische Charakteristika (die z.B. mit dem aktuellen Endgerät des Benutzers abgeglichen werden müssen) fließen hier ein. Hier gibt es sicherlich keine universelle Antwort, was wie in den Auswahlprozess einfließen soll. Stattdessen geht es wiederum darum, einen flexiblen Rahmen für eine Dienstimplementierung zu schaffen. Kern hierbei ist die Differenzierung in »harte« und »weiche« Bedingungen (vgl. [Sch03a]). Harte Bedingungen schließen nicht passende Ergebnisse aus der Ergebnismenge aus, während weiche Bedingungen dazu benutzt werden, den einzelnen Ergebnissen einen Grad der Übereinstimmung zuzuordnen. Lernziele sind dabei sicherlich harte Bedingungen, während Lerncharakteristika wie semantische Dichte oder Verspieltheit eher als weiche Faktoren einfließen.

In der Umsetzung werden harte Bedingungen als Selektionsausdruck in der Anfrage an die Artefaktverwaltung benutzt; weiche Bedingungen gehen in eine Bewertungsfunktion ein, die ein Übereinstimmungsmaß der Ergebniseinträge berechnet. Wenn man von einer Konjunktion von weichen Bedingungen $b_i := (a_i = v_i)$ ausgeht, wo a_i ein Metadatenattribut und v_i den präferierten Wert darstellt, so sieht eine Bewertungsfunktion für ein Ergebniselement e mit den Attributwerten $e[a_i]$ wie folgt aus:

$$B(e) := \frac{1}{n} \sum_{i=1 \dots n} w_i B_{a_i}(e[a_i], v_i)$$

wobei $0 \leq w_i \leq 1$ ein Gewichtungsfaktor und B_{a_i} eine attributspezifische Bewertungsfunktion darstellt. Je nach Attribut kommen hier Abstandsmaße (sowohl für metrische als auch ordinale Wertemengen) zum Einsatz. Als Erweiterung kommen auch taxonomiebasierte Maße zum Einsatz (vgl. [HZB⁺06]). In dieser Bewertungsfunktion fließt auch das

²Am Rande bemerkt sind auch viele im Bildungs- wie auch in anderen Personalbereichen verwendeten Kompetenzen eigentlich viel zu grobgranular, um beobachtet bzw. diagnostiziert oder aber vermittelt zu werden. Ein genauerer Blick entlarvt die Begriffsverwendung dann auch meist als nicht wohl definiert.

Prinzip der Einzigartigkeit vs. Redundanz aus Kapitel 5 ein, indem ein *degree-of-redundancy* aus der Anzahl der Kompetenzstufen, die neu erworben werden, und dem Erfüllungsgrad der Voraussetzungen berechnet wird. Die Abweichung von einem idealen Redundanzgrad fließt dann in die Bewertungsfunktion ein.

Mit diesen Überlegungen erhält man folgenden Algorithmus; dabei ist die Bewertungsfunktion *B* als *assess*-Methode repräsentiert, die jedes Element der Ergebnismenge mit einem Gewichtungsfaktor bewertet.

```
WeightedSet<LearningObject> getLearningObjects(  
    UserContext uc, Set<Competency> C)  
{  
    C = inferWholeFromParts(C);  
    ArtefactQuery q = 'LearningObject(?x),  
        has-objective(?x,?c), (c in C)';  
  
    q.append(getHardConditions(uc));  
  
    Set<LearningObject> A =  
        artefactRepository.getArtefacts(q);  
  
    WeightedSet<LearningObject> R = assess(A,uc);  
    return R;  
}
```

Auswahl von anderen Artefakten

Wie schon im letzten Kapitel skizziert, können andere Artefakte als Lernobjekt nur auf der Basis des Themas (Relation *covers*) behandelt werden, was freilich semantisch eine viel schwächere Aussagekraft liefert. Wissensartefakte dienen nicht der Vermittlung von Kompetenzen, sie können allenfalls – wie Suchaktivitäten im allgemeinen – Elemente eines Lernprozesses sein, über das jedoch das System keine weitere Anhaltspunkte hat. Stattdessen haben sie »Themen« (über die Relation *covers*), was neben Kompetenzen auch Organisationsentitäten umfassen kann (z.B. ein Dokument über einen Prozessschritt).

Zusätzlich kann auf eine evtl. vorhandene Reifeklassifikation zurückgegriffen werden. Denn ein Konstruktionsmerkmal des Wissensreifungsprozesses war die Vermittelbarkeit außerhalb des unmittelbaren Entstehungskontextes. Während Lernobjekte oder gar Kurse eine universelle Gültigkeit haben, sind Gebrauchsdokumente oder gar Artefakte aus informellen Kommunikationsprozessen nur in gewisser Nähe zu ihrem Entstehungskontext bzw. mit einer tieferen Kompetenzstufe im jeweiligen Themenbereich nützlich.

```
WeightedSet<InformationArtefact> getArtefacts(  
    UserContext uc, Set<Competency> C)
```

```

{
  Set<OrganizationEntity> o = uc.getValues('is-in');
  Set<Topic> topics = C.union(o);
  Set<Topic> topics =
    ontologyService.getAllNarrowerTopics(topics);

  ArtefactQuery q = 'InformationArtefact(?x),
    covers(?x,?t), (t in topics)';

  Set<InformationArtefact> A =
    artefactRepository.getArtefacts(q);

  WeightedSet<InformationArtefact> R = assess(A,uc);
  return R;
}

```

In die Bewertung (*assess*) sollte neben technischen Faktoren (ähnlich zum vorangegangenen Abschnitt) auch der Reifegrad einfließen: Sind die Reifegrade m_i, \dots, m_r und den Kompetenzniveaus n_1, \dots, n_s gegeben, so lässt sich für eine Ressource e mit Reifegrad m_i die Bewertungsfunktion wie folgt berechnen:

$$B_{maturity}(e) := 1 - \left| \left(1 - \frac{i}{r}\right) - \frac{j}{s} \right|$$

wobei n_j das Kompetenzniveau der nächsten übergeordneten Kompetenz des Benutzers ist (bzw. n_0 falls diese nicht vorhanden ist).

Auswahl von Personen

Auf den ersten Blick erscheint die Auswahl von Ansprechpartnern ein einfacher Spezialfall der Auswahl von Lernobjekten zu sein, da man Ansprechpartner anhand ihrer Kompetenzen beschreiben und damit auch auswählen kann. Und Expertensuchsysteme, wie sie im Wissensmanagement üblich sind, scheinen von der technischen Seite auch keiner großen Unterstützung mehr zu bedürfen. Doch informelle Lehr-Lernprozesse erfordern eine etwas komplexere Auswahllogik, was im wesentlichen darauf zurückzuführen ist, dass das jemanden Fragen das Eingeständnis, dass man selbst etwas nicht und jemand anders besser weiß, voraussetzt:

- Im Gegensatz zu klassischen Expertensuchsystemen ist für solche informelle Lehrsituationen nicht immer der Experte der am besten geeignetste. In den meisten Fällen reicht derjenige Kollege, der »etwas mehr« weiß aus, um viel effizienter das Benötigte zu vermitteln. Dies realisiert auch das Prinzip der Balance Einzigartigkeit vs. Redundanz aus Kapitel 5.

- Ob ein Ansprechpartner geeignet ist, hängt auch von der sozialen Beziehung zu ihm ab. Hier ist aus naheliegenden Gründen ein Kollege auf »der gleichen Stufe« besser geeignet als ein Vorgesetzter oder ein Unbekannter.

Wie bei den Lernobjekten bzw. Lernprogrammen muss hier also eine analoge Bewertungsfunktion *assess* zum Einsatz kommen, die einen Relevanzwert für mögliche informelle Lehrer liefert, wobei »mögliche« informelle Lehrer definiert sind als diejenigen, die über die erforderliche Kompetenz oder eine Kompetenz, die die gesuchte subsumiert, verfügen:

- Für den Aspekt der sozialen Beziehung ordnet man den unterschiedlichen Beziehungsmerkmalen Werte zu, so z.B. 0.5 für Vorgesetzte und 1 für Kollegen.
- Zur Bewertung des Kompetenzunterschiedes lässt sich die Summe über die Differenz der Kompetenzstufen für die erforderlichen Kompetenzen heranziehen, wobei die Summe so normiert wird, dass sie 1 ergibt, wenn exakt 1 Stufe Unterschied besteht, und der Bewertungsfaktor abnimmt für größere Unterschiede
- Ergänzend dazu kann man bei Ansprechpartnern auch nicht bei der bloßen statischen Empfehlung stehen bleiben, dass jemand ein geeigneter Ansprechpartner ist, sondern muss zudem den aktuellen Kontext auf der Lehrendenseite mit einbeziehen: Ist die Person verfügbar? Ist sie derzeit mutmaßlich dazu bereit, als informell Lehrender aufzutreten, was wiederum u.a. von der aktuellen Aufgabe und der Qualität der sozialen Beziehung abhängt. Hier wird auf das in [GBK06] eingeführte Maß des *degree of efficiency* zurückgegriffen.

Die Auswahl von Personen kann schließlich auch nicht nur über Kompetenzen, sondern direkt über den (Organisations-)Kontext erfolgen, wenn man Ansprechpartner empfehlen will, die kürzlich in demselben Teilkontext waren (innerhalb einer Zeitspanne T). Hierzu werden entsprechende Anfragen an den Kontextdienst gestellt, die Personen ermitteln, die innerhalb eines bestimmten Zeitraums dieselbe Aufgabe oder denselben Prozessschritt bearbeitet haben. Bewertungen bei mehreren Ergebnissen können analog zu den obigen Bewertungsverfahren erfolgen.

Mit diesen Überlegungen lässt sich das in Pseudo-Code-Notation wie folgt beschreiben, wobei analog zur Methode *getArtefacts* die Kompetenzlücke neben dem Benutzerkontext als Eingabe dient:

```
WeightedSet<Employee> getPersons(  
    UserContext u, Set<Competency> C)  
{  
    Set<OrganizationEntity> O = uc.getValues('is-in');
```

```
UserQuery q =
'SELECT USERS
VALID [NOW-T,NOW]
WHERE has-competency IN
  (SELECT ?c
   WHERE {?c subsumes ?d},
   {?d IN C} )
OR is-in IN O';

Set<Employee> P =
  userContextService.getUsers(q);

WeightedSet<Employee> R = assess(P,uc);
return R;
}
```

10.2. Lernprogramme zusammenstellen

Der zweite Problemkomplex besteht aus der Zusammenstellung von individuellen Lernobjekten zu sinnvollen Lernprogrammen (statt isolierten Lernobjekten) – also der didaktischen Aufbereitung durch Sequenzierung der Lernressourcen. Dies motiviert sich aus der Erkenntnis, dass für den Lernprozess ein ausgewogenes Verhältnis aus Redundanz und Einzigartigkeit (vgl. Abschnitt 5.3.1) förderlich ist (vgl. auch Anforderung AF4). Technisch ist dies aus zwei Gründen erforderlich:

- Eine Menge von Zielkompetenzen lässt sich oft nur durch mehrere Lernobjekte erreichen.
- Lernobjekte haben typischerweise Voraussetzungen und direkte Abhängigkeiten. In manchen Fällen ist es »sinnvoll«, in anderen »unumgänglich«, dass man sich zuvor bestimmtes Wissen angeeignet hat.

Gesucht ist daher ein Verfahren, das zu einer Menge von Zielkompetenzen ein Lernprogramm zusammenstellt, innerhalb dessen die Lernobjekte so geordnet sind, dass alle Abhängigkeiten berücksichtigt sind, insbesondere dass die Voraussetzungen von Lernobjekten entweder bereits durch den Lernenden beherrscht werden oder vorausgehende Lernobjekte vermittelt werden. Hierbei gilt es als Randbedingungen zu beachten:

- Die Erstellung des Lernprogrammes muss vollautomatisch und innerhalb kurzer Reaktionszeiten erfolgen.

- Bei der Vermittlung von Kompetenzen und der Berücksichtigung von Voraussetzungen ist die Subsumtionsordnung auf Kompetenzen gemäß dem vorangegangenen Abschnitt zu berücksichtigen.

10.2.1. Stand der Technik: Semantische Komposition von Lernprogrammen

BRUSILOVSKY & VASSILEVA [BV03] unterscheiden bei der Komposition von Lernprogrammen die beiden Alternativen einer adaptiven Kursgenerierung (*adaptive courseware generation*), bei der ein Lernprogramm bzw. Kurs vor Beginn der Lernaktivitäten generiert werden, und einer dynamischen Kursgenerierung, bei der die Navigationsstrukturen zur Laufzeit auf der Basis der Aktionen und Ergebnisse des Lernenden angepaßt werden. Hierbei bezeichnen beide Formen einen Prozess, der für die Zielgruppe Lernobjekte auswählt und sie zielgruppengerecht anordnet (in der Regel sequentiell) [Kno03]. Für die vorliegende Arbeit steht dabei primär die adaptive Generierung im Vordergrund.

Ein weiteres Unterscheidungsmerkmal ist der Automatisierungsgrad im Kompositionsprozess. Für stark pädagogisch geprägte Umgebungen wie z.B. schulisches oder universitäres Lernen ist nur ein Ansatz angemessen, der das pädagogische Handeln des Lehrenden unterstützt, anstatt es zu ersetzen versuchen. Hier kommen nur semiautomatische Verfahren in Betracht, denn vollautomatische Ansätze können durch ihre unvermeidliche Vereinfachung nicht der vollen Bandbreite des pädagogischen Handelns gerecht werden. Dementsprechend stehen dort Engineering-Fragestellungen im Vordergrund (vgl. [AL06], [DWC05], [FSD02], [BLRT03]) wie z.B. Vorgehensmodelle, didaktische Vorlagen und Wiederverwendung von Inhalten mit minimalem Anpassungsaufwand. Auch ist dort ein Fokus auf große Kurse als Zielobjekt zu verzeichnen.

Demgegenüber ist für die vorliegende Problemstellung eines Lernens bei Bedarf nur ein vollautomatischer Ansatz sinnvoll, auch wenn dieser aus pädagogischer Sicht sicherlich qualitativ nicht so hochwertige Ergebnisse liefert. Auch hier existieren zahlreiche Ansätze, die auf semantische Beziehungen zwischen den zu vermittelnden Domänenkonzepten aufbauen (vgl. z.B. [OBCW06], [KS04], [KS05a]). Was hier allerdings fehlt, ist eine Differenzierung nach unterschiedlichen Kompetenzniveaus oder die Differenzierung von Generalisierung und Komposition. Ebenso wird nur wenig auf andere Kontextfaktoren eingegangen.

10.2.2. Verfahren zur Zusammenstellung von Lernprogrammen

Deshalb stellen wir im folgenden ein Verfahren vor, das die obigen Anforderungen erfüllt, wobei grundsätzlich davon ausgegangen wird, dass zu generierende Lernprogramme vergleichsweise kurz sind (auch im Gegensatz zu den meisten existierenden Ansätzen). Ausgangspunkt für das Verfahren ist eine Menge von Kompetenzen Z , die als Zielkompe-

tenzen dienen, und die Menge M der aktuellen Kompetenzen des Mitarbeiters. Die Zielkompetenzen sind dabei im Normalfall eine Untermenge der ermittelten Kompetenzlücke für die jeweilige Situation (wie im vorangegangenen Abschnitt beschrieben). Gesucht sind dann Sequenzen von Lernobjekten (lo_1, \dots, lo_n) , so dass gilt:

- $Z \ominus (M \cup \bigcup_{i=1..n} objectives(lo_i)) = \emptyset$, d.h. die Sequenz soll dem Mitarbeiter alle Kompetenzen vermitteln, sofern er über diese noch nicht verfügt. Falls dies nicht möglich ist, soll allerdings auch ein partielles Ergebnis zurückgeliefert werden.
- $prerequisites(lo_1) \ominus M = \emptyset$, d.h. der Lernende soll alle Voraussetzungen für das erste Lernobjekt erfüllen.
- $prerequisites(lo_i) \ominus (M \cup \bigcup_{j=1..i-1} objectives(lo_j)) = \emptyset$, d.h. das jeweils nächste Lernobjekt soll auch nur den bislang vermittelten bzw. vorhandenen Kompetenzstand erfordern.

Dies führt nicht zuletzt durch die Berücksichtigung von Abhängigkeiten (sowohl direkter als auch indirekter Art) bei der Zusammenstellung zu einem Optimierungsproblem. Letztendlich verbirgt sich dahinter eine Variante des Set-Cover-Problems [Vaz01], das bereits in seiner Reinform NP-vollständig ist. Deshalb wird für die Zusammenstellung der Lernobjekte ein approximatives Verfahren benutzt, das auf der Best-First-Suchstrategie basiert, da sich hierüber leicht die oben erwähnten »weichen« Kriterien einbinden lassen.

```

/* Vorbedingung: die Menge der zu vermittelnden Kompetenzen c enthält
keine Kompetenzen, die von einem der Lernobjekte des bereits
berechneten Lernprogrammfragments p vermittelt wird. */
List<LearningObject> getLearningProgram(UserContext u,
                                        List<Competency> c)
{
    // hole die Kompetenzen des Benutzers
    List<Competency> uc = getCompetencies(u);

    Set<LearningObject> p;
    do
    {
        WeightedList<LearningObject> L = getLearningObjects(u, c);

        // sortiere entsprechend der Bewertung
        L = sortWeightedList(L);

        // nehme das am besten bewertete Lernobjekte
        LearningObject l = L.removeFirst();
    }
}

```

```

Set<LearningObject> p' = { l }

// fügt transitiv alle Abhängigkeiten hinzu
p'.prepend(getAllDirectDependencies(l));

p.append(p');

/* berechne die Lücke zwischen
- den Zielkompetenzen und den Voraussetzungen des gewählten
  Lernobjektes auf der einen Seite
- und den Lernzielen des gewählten Objektes, den bereits die
  durch die Lernobjekte im Lernprogramm enthaltenen Lernziele
  (inkl. den gerade ausgewählten) sowie den aktuellen
  Kompetenzen des Benutzers auf der anderen Seite */
c = gap( c.union(p'.getPrerequisites(),
               p.getObjectives().union(uc) ) );
} while L is not empty

/* topologische Sortierung entsprechend der partiellen
  Ordnung */
return orderLO(p);
}

```

Der vorgestellte Algorithmus terminiert immer, auch wenn keine Lösung existiert, da die Schleife maximal so viele Durchläufe hat wie die Anzahl der zur Verfügung stehenden Lernobjekte. Durch die Ordnung der zu berücksichtigenden Lernobjekte nach der Gewichtung der Lernobjekte (Methode *sortWeightedList*) in der Ergebnisliste wird berücksichtigt, dass neben den reinen Lernzielen auch die »weichen« Faktoren die Angemessenheit eines Lernprogrammes wie oben betrachtet einbezogen werden.

Ordnung auf Lernobjekten

Für die topologische Sortierung der ausgewählten Lernobjekte in ein sequentielles Lernprogramm in der Methode *orderLO* wird eine partielle Ordnung benötigt, die die entsprechenden Abhängigkeiten berücksichtigt. Diese sei wie folgt definiert:

Definition 10.3 (Ordnung auf Lernobjekten) Für zwei Lernobjekte l_1 und l_2 gilt $l_1 \triangleleft l_2$ (d.h. l_1 kommt vor l_2), wenn mindestens eine der folgenden Bedingungen erfüllt ist:

- l_2 depends-on l_1
- $\text{prerequisites}(l_2) \cap \text{objectives}(l_1) \neq \emptyset$
- $\exists l_3 : l_1 \triangleleft l_3 \wedge l_3 \triangleleft l_2$

Damit die so definierte Relation auch eine strikte partielle Ordnung ergibt, muss zusätzlich gefordert werden, dass die Kombination aus direkten und indirekten Abhängigkeiten azyklisch ist. Dies muss bei der Erstellung der Lernobjekte als Konsistenzbedingung durchgesetzt werden.

10.3. Lernkoordination

Die bislang vorgestellte Dienstfunktionen sind rein passiv; sie erlauben die Ermittlung von Empfehlungen für eine bestimmte Situation des Benutzers. Auch wenn dies nicht im weiteren Fokus der Arbeit steht, werden für ein Lernunterstützungssystem als Ganzes auch Strategien benötigt, die festlegen, wann welche Intervention in den Lernprozess sinnvoll ist. Hier sind pädagogische Strategien gefragt, die ein automatisiertes Scaffolding realisieren können, indem sie gezielt Lernprozesse anstoßen. Insbesondere sind folgende Probleme zu adressieren:

- Gerade für einen neuen Mitarbeiter können sich in einer konkreten Arbeitssituation zahlreiche Anforderungen ergeben, die noch nicht durch seine Kompetenzen abgedeckt sind. Hier ist es sicherlich nicht im Sinne des arbeitsbegleitenden Lernens, dass die angebotenen Lernprogramme zu umfangreich sind. Hier muss eine pädagogische Strategie differenzieren zwischen dringenderen und weniger dringenden Kompetenzanforderungen, was beispielsweise darüber geschehen kann, dass typischerweise Anforderungen, die sich aus der Aufgabe ergeben, akutere Probleme adressieren, als Anforderungen, die sich aus der Rolle oder der Abteilungszugehörigkeit ergeben.
- Ob und in welcher Form Lerngelegenheiten empfohlen werden, ist eine weitere Entscheidung, die zu treffen ist. Hier ist vor allem eine Balance zu finden zwischen der Unterbrechung des Arbeitsflusses und des zu erwartenden Nutzens für den Lernenden.

10.4. Fazit

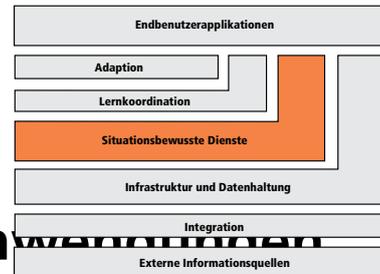
In diesem Kapitel wurde die Realisierungsmöglichkeiten für die in der Referenzarchitektur enthaltenen lernunterstützenden situationsbewussten Dienste thematisiert. Dabei stand nicht so sehr *eine* definitive Implementierung der jeweiligen Dienste im Vordergrund, sondern vielmehr ein Rahmen für unterschiedliche Implementierungslösungen. Die Spielräume liegen dabei vor allem in der Wahl der Bewertungsfunktionen (die in den Algorithmenfragmenten als *assess* enthalten ist). Durch diese Spielräume lassen sich Anpassungen und Feinjustierungen vornehmen, die für den jeweiligen Zielkontext erforderlich sind. Ebenso

stellen Sie die Stellschrauben dar, um eine kontinuierliche Verbesserung auf der Basis von Erfolgsmessung und Nutzerfeedback durchführen zu können.

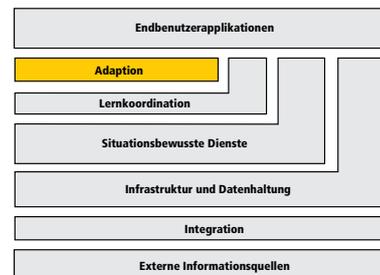
Die hier vorgestellte Lösung basiert auf der Annahme, dass sich die Leistungsanforderungen in einer bestimmten Arbeitssituation sowie die Ergebnisse von On-Demand-Lernaktivitäten durch feingranulare Kompetenzen beschreiben lassen. Dabei bleibt der konkrete Ansatz zur Modellierung von Kompetenzen offen (hier können unterschiedliche psychologische oder pädagogische Modelle zum Einsatz kommen – es wird nur ein Metavokabular bereitgestellt. Die durch die Feingranularität unvermeidbar umfangreichen Kompetenzkataloge werden durch die Berücksichtigung von Kompetenzbeziehungen besser handhabbar, da hierdurch Probleme wie die Spezifikation von Anforderungen und Kompetenzprofilen auf unterschiedlichen Abstraktionsebenen abgemildert werden.

11.

Adaption von Endbenutzeranwendungen



Damit dem Benutzer Lerngelegenheiten empfohlen und angezeigt werden können, werden neben den entsprechenden Diensten aus dem vorangegangenen Kapitel auch Endbenutzeranwendungen benötigt. Solche Anwendungen können viele Gestalten annehmen, so dass es in diesem Kapitel nicht um eine universelle Benutzerschnittstelle geht, sondern vielmehr um die flexible Anbindung an die wiederverwendbare Infrastruktur, d.h. um die Adaptionsschicht in der Referenzarchitektur (die Anforderung AN1 mit realisiert), die die Brücke zu den Anwendungen darstellt, mit denen der Anwender zumindest mittelbar in Kontakt kommt. Die folgenden Fragen stehen im Zentrum:



- *Wie können dynamisch zusammengestellte Lernprogramme in standardisierten Lernumgebungen angezeigt werden?* Gerade wenn es sich um pädagogisch aufbereitete Lerninhalte handelt, sind diese typischerweise nicht als einfache Dokumente (z.B. PDF) realisiert, sondern als interaktive Inhalte, die eine spezielle Ablaufumgebung benötigen. Hier muss eine Integration stattfinden.
- *Wie können Lernobjekten Informationen über den Kontext innerhalb von standardisierten Lernumgebungen so bereitgestellt werden, dass möglichst wenig Veränderungen notwendig sind (AF6)?* Über das reine Anzeigen hinaus sollen Lernobjekte auch in der Lage sein, auf den Kontext des Benutzers einzugehen.
- *Wie können dem Benutzer Lernempfehlungen zur Initiierung eines Lernprozesses präsentiert werden (AF3)?* Hierzu gilt es, in der Umgebung des Benutzers entsprechende Benachrichtigungsmechanismen vorzusehen.

11.1. Integration in Lernumgebungen

Ein Aspekt der Integration ist die Anzeige von Lernressourcen. Hierbei geht es vor allem um die Anzeige von Lernobjekten in einer geeigneten Lernumgebungen, um dort die jeweilige Funktionalität (wie z.B. Assessment und Integration mit anderen Unternehmenssystemen) zu nutzen.

11.1.1. SCORM-kompatible Lernmanagementsysteme (LMS)

Der derzeitige De-Facto-Standard für Lernmanagementsysteme und Inhalte, die herstellerübergreifend zwischen Systemen ausgetauscht werden ist *SCORM* (*Sharable Content Object Reference Model*) der *Advanced Distributed Learning Initiative* (*ADL*), das ein Referenzmodell für webbasierte Lernmanagementsysteme und technische Umsetzungen (z.B. als XML-Formate oder auch Referenzimplementierung des Standards) bereitstellt. Der Kern von SCORM sind wiederverwendbare Lernobjekte, die als »Sharable Content Objects« (SCO) bezeichnet werden, die mittels webkonformen Techniken erstellt werden und so verpackt werden können, dass sie in jeder SCORM-Umgebungen (Runtime Environment, RTE) lauffähig sind. Für die Kommunikation zwischen SCO und Lernumgebung wird eine standardisierte API definiert (z.B. zur Rückmeldung des Lernerfolges).

Für das Problem der Bereitstellung von Kontextinformationen für Lernobjekte sind besonders die folgenden Teile von SCORM relevant:

- **Content Aggregation Model (CAM).** CAM beschreibt die innere Struktur von Lernobjekten und wie der Inhalt technisch zu »verpacken« ist. Kern der Spezifikation ist das Paketformat und dessen Manifestformat, indem sowohl die beschreibenden Metadaten (IEEE LOM) als auch die Strukturmetadaten enthalten sind. Wenn wir die dynamisch generierten Lernprogramme konform zu diesem Modell repräsentieren können, ist sichergestellt, dass SCORM-kompatible Lernumgebungen diese auch problemlos verarbeiten können.
- **SCORM-Laufzeitumgebung (RTE).** Die RTE beschreibt die Anforderungen an eine Ablaufumgebung, die SCORM-konforme Lernobjekte abspielen will. Dies umfasst eine Definition der Kommunikation zwischen Inhalten und dem LMS sowohl auf Protokoll als auch Schemaebene der ausgetauschten Informationen. Den Lernobjekten (bzw. SCOs) wird eine API (auf der Basis von JavaScript) angeboten, über die die Kommunikation technisch abgewickelt wird. Für die Bereitstellung der Kontextinformationen ist diese Schnittstelle zu verbreitern.

In den folgenden beiden Unterabschnitten werden zunächst die jeweilig relevanten Teile von SCORM noch genauer vorgestellt, getrennt nach den jeweiligen Problemen. Zunächst

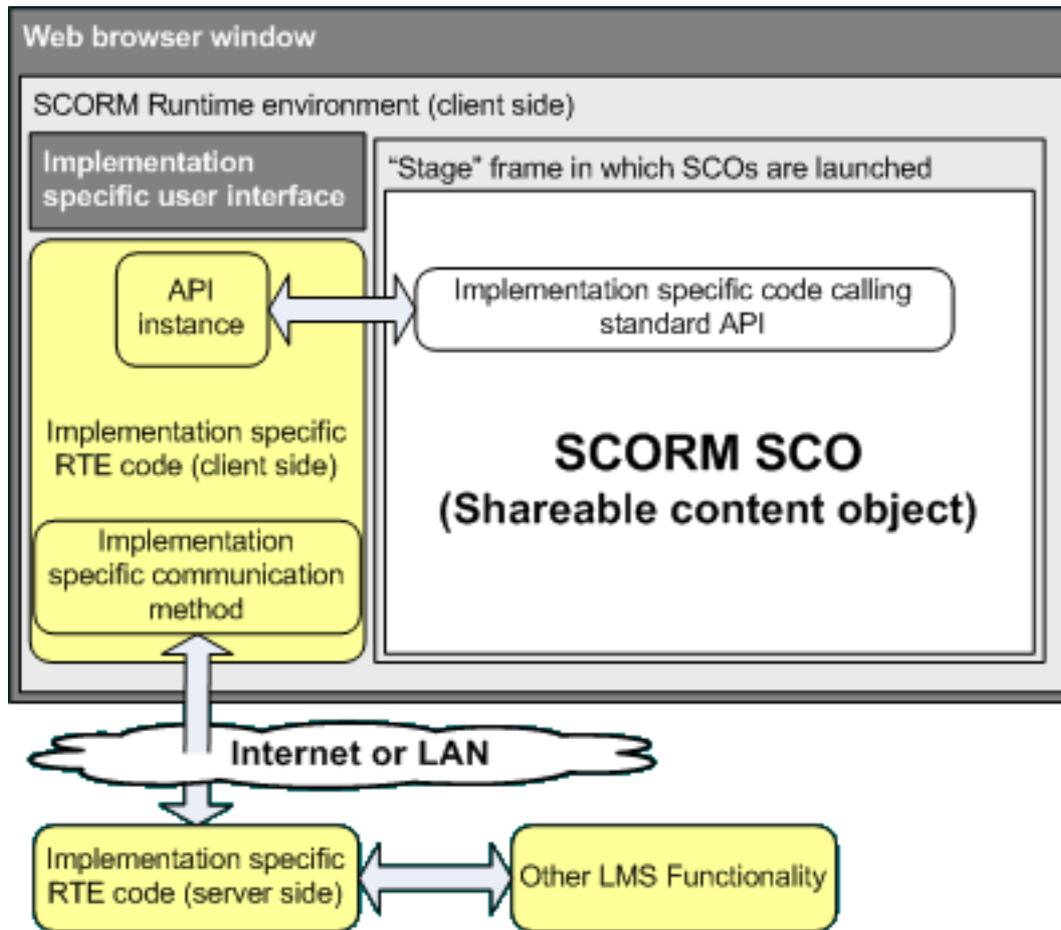


Abbildung 11.1.: Grundkonzept eines webbasiertes Lernsystems [Ost05]

widmen wir uns den dynamisch generierten Lernprogrammen, bevor wir die Bereitstellung von Kontextinformationen betrachten.

11.1.2. Ausführen von generierten Lernprogrammen

Das Paketformat des *SCORM Content Aggregation Models* (CAM) [Adv04] basiert (analog zu den Paketformaten der Java-Welt wie JAR, EAR o.ä.) auf ZIP-komprimierten Archiven (als *Content Package* bezeichnet, vgl. Abb. 11.2), die eine Metadatendatei (als »Manifest« bezeichnet) enthalten. Dieses Manifest ist eine XML-Datei, die zum einen Metadaten über das gesamte Paket enthält; hierzu wird der LOM-Standard [Lea02] (mit abgeschwächten Kardinalitäten durch mehr optionale Attribute) verwendet. Zum anderen beschreibt es aber

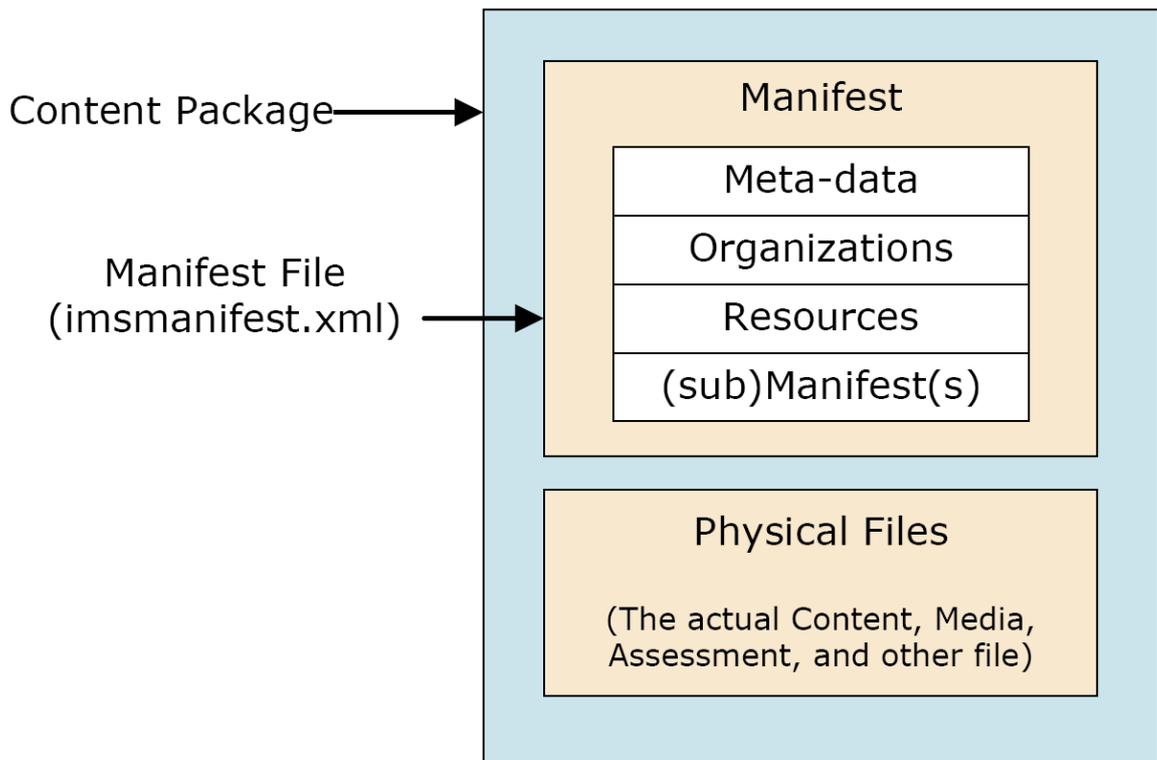


Abbildung 11.2.: Struktur eines SCORM-Paketes [Adv04]

auch die einzelnen Teile des Paketes. So kann ein SCO wiederum aus mehreren SCOs oder aus atomaren »Assets« (z.B. Bilder, HTML-Seiten) bestehen; diese werden im sog. »Resources«-Teil beschrieben. Hierbei können die SCOs oder Assets sowohl eingebettet sein in das Paket, als auch extern über eine URL referenziert werden. Über den »Organization« genannten Teil lassen sich dann Lernpfade angeben, die die eine sinnvolle Navigationsstruktur zwischen den Assets/SCOs festlegen (und insbesondere auch den Einstiegspunkt definieren).

Das CAM erlaubt relativ einfach die Einbindung von dynamisch generierten Paketen – sogar ohne die Notwendigkeit, ein physisches Paket zu erstellen. Im SCORM-Jargon entsprechen den Lernobjekten dieser Arbeit die sog. Sharable Content Objects (SCO), die sich hierarchisch zusammensetzen lassen. Jeweils beschreibt eine XML-basierte Manifest-Datei die Komposition. So ist es möglich, dass bei der Lernprogrammgenerierung einfach eine solche Manifest-Datei generiert wird, die die jeweiligen Lernobjekte als entfernte Referenz enthalten. Dies lässt sich am besten schematisch anhand einer Manifest-Datei illustrieren:

```

<manifest>
...
<organizations default="X">
  <organization identifier="X">
    <title>[Learning Program Title]</title>
    <item identifier="L01" isvisible="true"
      identifierref="R_L01" />
    <item identifier="L02" isvisible="true"
      identifierref="R_L02" />
    <metadata>
      <schema>ADL SCORM</schema>
      <schemaversion>1.2</schemaversion>
      <adlcp:location>lpmetadata.xml</adlcp:location>
    </metadata>
  </organization>
</organizations>
<resources>
  <resource identifier="R_L01" type="webcontent"
    adlcp:scormtype="sco"
    href="[remoteURL1]" />
  <resource identifier="R_L02" type="webcontent"
    adlcp:scormtype="sco"
    href="[remoteURL2]" />
</resources>
</manifest>

```

Damit muss die Liste der Lernobjekte, die vom Dienst für die Lernprogrammzusammensetzung geliefert, nur in dieses Manifest-Format transformiert werden und die Lernprogramme können in SCORM-Lernumgebungen angezeigt und verarbeitet werden. Durch die externe Einbindung der Lernobjekte kann die Lernumgebung den Inhalt direkt von der Artefaktverwaltung anfordern.

11.1.3. Kontextbewusste Lernobjekte: Bereitstellung von Kontextinformation für Lernobjekte

In Anforderung *AF6* wurde gefordert, dass das System zur besseren Situierung von Lerngelegenheiten auch Lernobjekten Kontextinformationen bereitstellen muss. Dies läuft auf das Konzept von **kontextbewussten Lernobjekten** hinaus (vgl. Abb. 11.3), die über Anpassungsmöglichkeiten an den Kontext des Lernenden verfügen. Derzeit sieht SCORMs Laufzeitumgebung vor, dass Lernobjekte mit dem Lernmanagementsystem mittels JavaScript-Funktionen über einen SCORM-API-Adapter (oft als LMS-spezifisches Java-Applet übertragen) kommunizieren können. Dabei definiert die Spezifikation der

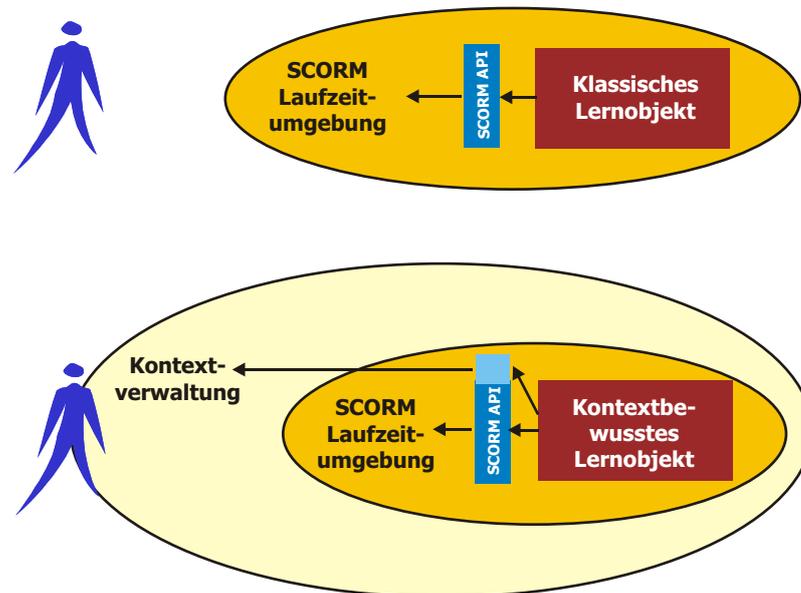


Abbildung 11.3.: Klassische und kontextbewusste Lernobjekte aus SCORM-Sicht

Laufzeitumgebung das sog. CMI-Schema (*computer managed instruction*) für den Austausch von Daten zwischen Lernobjekt und Lernmanagementsystem. Der Hauptzweck ist hierbei derzeit die persistente Speicherung von Laufzeitinformationen wie z.B. der erreichten Punktzahl oder der aktuelle Fortschritt über Sitzungen hinweg, so dass eine nahtlose Wiederaufnahme der Lernaktivität ermöglicht wird. Die Interaktion mit dem Server geschieht über zwei API-Methoden *LMSSetValue(String element, String value)* und *LMSGetValue(String element)*. Die verfügbaren Elemente sind im CMI standardisiert und werden über Pseudo-Pfadausdrücke als *datamodel.group.element* dargestellt, wobei *datamodel* derzeit stets *cmi* ist. Bei kontextbewussten Lernobjekten will man diesen Kommunikationskanal verbreiten, so dass nicht nur interne Zustandsdaten gespeichert und geladen werden können, sondern auch die Möglichkeit geschaffen wird, auf Informationen über den Kontext des Benutzers zuzugreifen, der extern (also außerhalb des Lernobjektes und auch der Lernplattform) gespeichert wird.

Da die API-Architektur auf eine beliebige Erweiterung der ausgetauschten Daten prinzipiell vorbereitet ist, wird einfach ein neuer Namensraum eingeführt. Um auch hier auf mehrere Kontextschemata (vgl. Teil 3) vorbereitet zu sein, werden Identifikatoren mit der Struktur *context.[schemaURI].[uri]* eingesetzt, wobei *schemaURI* das verwendete Kontextschema und *uri* das Kontextmerkmal eindeutig identifiziert. Dabei sind auch mehrwertige und komplexe Merkmale kein Problem, da die SCORM Laufzeitumgebung bereits einen

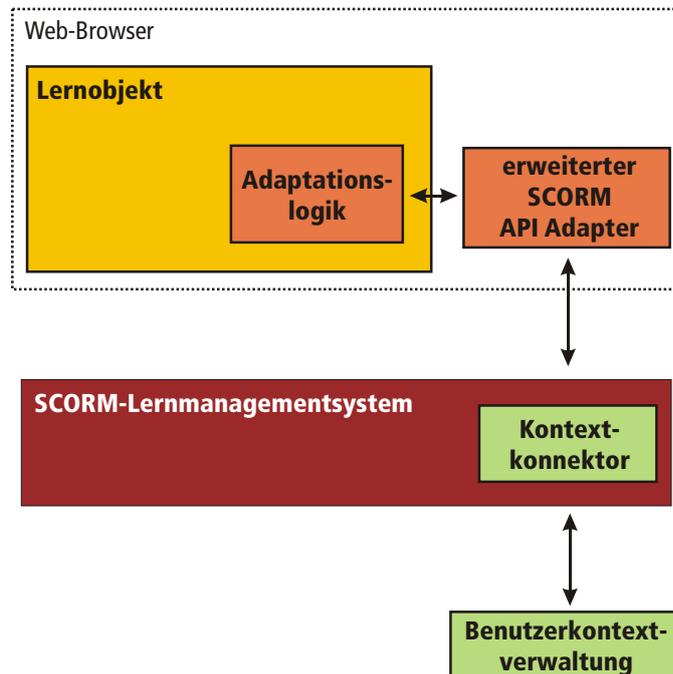


Abbildung 11.4.: Umsetzung kontextbewusster Lernobjekte mit SCORM-kompatiblen LMS

Mechanismus hierfür vorsieht, der einfach übernommen werden kann.

Um eine SCORM-kompatible Lernumgebung nun in die Lage zu versetzen, kontextbewusste Lernobjekte zu unterstützen, müssen die folgenden beiden Punkte umgesetzt werden (vgl. Abb. 11.4).

- **Erweiterung des SCORM-API-Adapters.** Der SCORM-API-Adapter ist die Brücke zwischen der Lernobjekt-Sandbox und der Umgebung. Konzeptionell sollte das Hinzufügen eines weiteren Datenmodells transparent für den API-Adapter sein, jedoch kann die Einführung komplexerer Datenstrukturen dazu führen, dass das Interaktionsprotokoll mit dem Server überarbeitet werden muss. Beispielsweise materialisieren API-Adapter-Implementierungen oft CMI-Ausprägungen, um die Kommunikation mit dem Server zu minimieren, während große Kontextmodelle und dynamische Umgebungen (wie hier betrachtet) es erforderlich machen, zu einem virtuellen Ansatz zu greifen, bei dem die angeforderten Kontextmerkmale bei Bedarf geholt werden.
- **Hinzufügen eines Kontextkonnektors zum LMS.** Die wichtigste Änderung ist auf Seiten des LMS die Erweiterung um eine Kontextkonnektorkomponente, der aus

SCORM-API-Aufrufen *LMSGetValue* und *LMSSetValue* Aufrufe an den Kontextmanager macht.

Im Gegensatz zu anderen Ansätzen, Adaptivität in den SCORM-Standard zu verlagern (z.B. [MGBG04] oder [RLFVDRPA06]), verlagert der Ansatz der kontextbewussten Lernobjekte Adaptionlogik vollständig in die Lernobjekte, da mit zunehmender Kontextualisierungstiefe sich die Adaptionstrategien kaum noch generisch spezifizieren lassen. Deskriptive Annotation funktioniert zwar recht gut, wenn sich beispielsweise Navigationsstruktur anhand einfacher Bedingungen an den Fortschritt des Lernenden anpassen; sie geraten bei komplexerer Adaptionlogik (z.B. bei komplexen Zusammenhängen, die von mehreren Kontextmerkmalen abhängen) schnell an ihre Grenzen. Vorteil hiervon ist auch, dass Lernmanagementsysteme (und die sie spezifizierenden Standards) von der Adaptionaufgabe entlastet werden. So kann Adaption auch »minimalinvasiv« in Standard-System Einzug halten (wie von Anforderung *AF6* gefordert).

11.2. Lernassistent: Anzeige von Lernempfehlungen

Die Hauptaufgabe des Lernassistenten ist die Intervention in die Arbeitsprozesse, die der Initiierung und Steuerung von Lernprozessen dienen. Grundsätzlich lassen sich die Interventionen in zwei Grundalternativen unterteilen: (pro-)aktive und passive Präsentation. Bei der passiven Präsentation bietet das System auf Anfrage des Benutzers Aktionen an (was in seiner Extremform ein einfaches »Suchfeld« sein kann), bei der proaktiven Präsentation werden Empfehlungen angezeigt, auf die der Benutzer hingewiesen wird (was in seiner Extremform ein »Zwingen« zum Lernen sein kann).

Für eine Integration von Arbeitsprozessen und Lernprozessen muss dem Benutzer die Freiheit gegeben werden, ob er weiterarbeiten oder vom System empfohlene Lernprozesse beginnen will. Deshalb eignen sich besonders periphere Anzeigetechniken, die eine Balance anstreben zwischen Unterstützungsangebot und potentieller Störung durch Unterbrechung. Aktuelle Desktopumgebungen bieten hier unterschiedliche Möglichkeiten an (vgl. Abb. 11.5):

1. Anzeige im Traybereich
 - a) durch ein animiertes Icon
 - b) durch Hinweisblasen
2. Popup-Fenster (u.U. semi-transparent)
3. Permanent sichtbare Seitenleiste mit aktualisierten Inhalten (z.B. Google Desktop Sidebar, Windows Vista Sidebar)



Abbildung 11.5.: Konzeptuelle Übersicht über die möglichen Realisierungsformen eines Lernassistenten

4. Direkte Integration in die (Fach-)Anwendung

An dieser Stelle geht es dabei nicht um die konkrete Implementierung solcher Benachrichtigungen des Benutzers, sondern um die Bereitstellung wiederverwendbarer Komponenten bzw. Strategien, um diese umzusetzen, insbesondere wenn dies eine Integration in bestehende Anwendungen bedeutet, die dann clientseitig den Lernassistenten repräsentieren.

Für die Integration können hierbei zwei unterschiedliche Integrationsebenen gewählt werden: die Informationsebene und die Präsentationsebene. Auf der *Informationsebene* (wie sie sich z.B. für die Integration in die Fachanwendung anbieten kann) »kennt« die benachrichtigende Anwendung die Informationsobjekte, also die Liste von empfohlenen Lerngelegenheiten und ihre Aufrufmöglichkeiten. Sie erhält sie beispielsweise als XML-Datenstruktur und visualisiert sie an der Benutzerschnittstelle. Dies eröffnet die Möglichkeit einer tieferen Integration in die jeweilige Benutzerschnittstelle, erfordert allerdings auch jeweils eigene Anpassungen. Auf der Präsentationsebene wählt man ein universelles Darstellungsformat (wie es z.B. in Form von HTML bereits existiert), in dem die Empfehlungen repräsentiert werden, so dass nur noch ein Durchreichen an die Anzeige Komponente erforderlich ist. Die darstellende Anwendung muss also nicht wissen, welche Art von Informationen angezeigt werden.

Konsequenz hieraus ist, dass für die Integration auf Informationsebene die Lernkoordination und die Schicht der situationsbewussten Dienste direkt genutzt werden kann. Für die Integration auf Präsentationsebene ist allerdings eine generische Komponente (der serverseitige Lernassistent) sinnvoll, die das Darstellungsformat generiert. Hiermit lassen sich dann z.B. unterschiedliche Sidebar-Technologien mit geringem Aufwand unterstützen.

Wie interagiert nun dieser serverseitige Lernassistent mit dem clientseitigen Lernassistenten in diesem Fall? In Abb. 11.6 ist ein exemplarischer Ablauf dargestellt; der clientseitige Teil wird dabei als »DesktopNotifier«, der serverseitige Teil des Lernassistenten als »LearningAssistantServlet« bezeichnet. Im ersten Schritt registriert sich der *DesktopNotifier* beim *LearningAssistantServlet*, um sich zu authentifizieren. Dies veranlasst eine Registrierung beim Lernkoordinator. Liegen neue Empfehlungen vor, benachrichtigt der Lernkoordinator das *LearningAssistantServlet*, das die Empfehlungen rendert und zwischenspeichert, bis sie vom *DesktopNotifier* abgerufen werden.

Exemplarisch sei in Abb. 11.7 gezeigt, wie eine solche Lösung für die Integration in Google Desktop aussehen kann. Hierbei wird ein (beinahe) vollständiger Auszug aus der Haupt-XML-Datei gezeigt.

Beim Öffnen wird dabei die Authentifizierungsinformation übertragen. Im Beispiel wird anschließend alle 10 Sekunden die Liste mit den Empfehlungen aktualisiert.

11.3. Fazit

In diesem Kapitel wurden ausgewählte Fragen zur Integration mit Anwendungen betrachtet. Dies betraf insbesondere die Integration mit traditionellen, auf dem Branchenstandard SCORM basierenden Lernmanagementsystemen. Hier wurde zum einen betrachtet, wie die generierten Lernprogramme in SCORM-kompatiblen Systemen angezeigt werden kön-

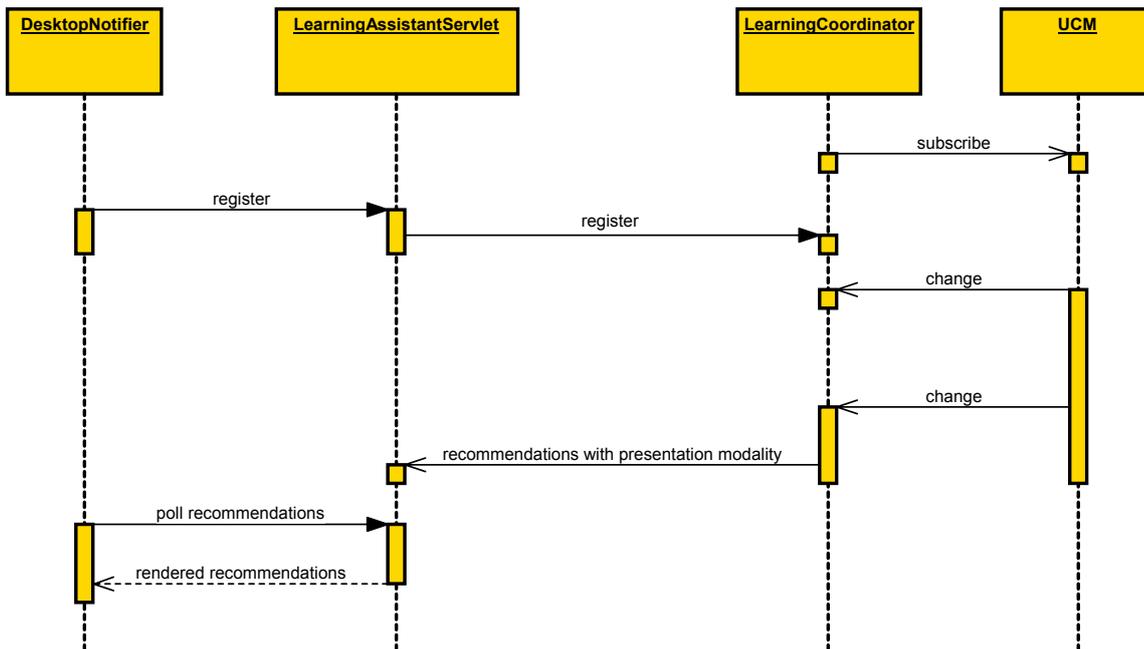


Abbildung 11.6.: Interaktionsdiagramm zwischen Benachrichtigungskomponente und dem Lernkoordinator

nen. Zum anderen wurde das Konzept der kontextbewussten Lernobjekte eingeführt, um Lernobjekten so die Möglichkeit zu verschaffen, auf Kontextinformationen zurückzugreifen.

Mit diesen Betrachtungen ist der Teil der lernunterstützenden Mehrwertdiensten abgeschlossen. Es blieben dabei einige Bereiche ausgeklammert, weil sie sich nur schlecht generisch, das heißt unabhängig von der konkreten Anwendung, im Rahmen des Domain Engineering realisieren lassen. Dies betrifft vor allem die Lernkoordination, aber auch die eigentliche Benutzerschnittstelle.

Bislang stand im Vordergrund, welchen Mehrwert für den Lernenden man mit welchen Kontextinformationen erzielen kann – unter der Voraussetzung, dass man über diese Informationen auch verfügt. Im folgenden Teil soll es nun noch um eine Infrastruktur gehen, die tatsächlich in der Lage ist, diese Informationen mit Hilfe von heuristischen Auswertungen des Benutzerverhaltens zu ermitteln.

```
<view width="200" height="200" onopen="view_onopen()">
<script language="jscript">
  function view_onopen()
  {
    // Authentifizierung
    var request = new XMLHttpRequest();
    var url = '[authentication url]';
    request.open('GET', url, true);
    request.send();

    // periodische Aktualisierung
    setInterval( "refresh()", 10000 );
  }

  function refresh()
  {
    // asynchrones Laden der HTML-Seite mit den Empfehlungen
    var request = new XMLHttpRequest();
    request.onreadystatechange = newData;

    var url = '[url to the learning assistant]';
    request.open('GET', url, true);
    request.send();
  }

  function newData()
  {
    // Anzeige der Empfehlungen
    recommendations.innerHTML = request.responseText;
  }
</script>
// container für die Anzeige
<div background="#FFFFFF" width="200" height="200"
  id='recommendations' />
</view>
```

Abbildung 11.7.: Integration in Google Desktop (Haupt-XML-Datei)

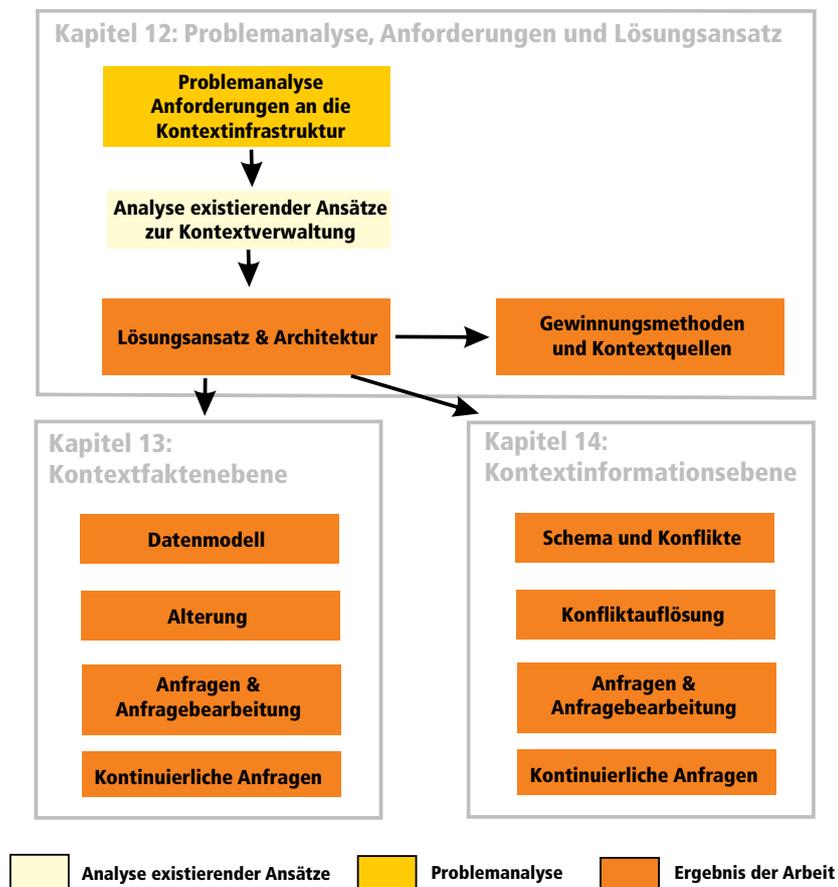
Teil IV.

Kontextinfrastruktur

Problemstellung

Grundlage für die situationsbewussten Dienste ist eine Infrastruktur, die ein Abbild der Situation des Benutzers technisch bereitstellt. Hierbei sind besonders die unvollkommenen Ermittlungsmethoden zu berücksichtigen.

Vorgehensweise



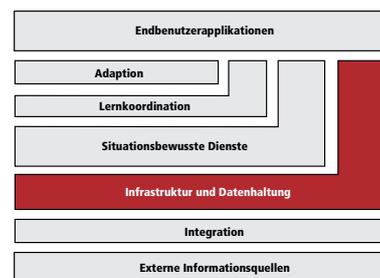
12.

Problemanalyse, Anforderungen und Lösungsansatz

Im vorangegangenen Teil wurde eine dienstorientierte Infrastruktur erarbeitet, die es erlaubt, das situationsgesteuerte Lernen informationstechnisch zu unterstützen. Hierfür waren Kontextmanagement und die ihr zugrundeliegenden Kontextquellen essentiell, da sie die erforderlichen Informationen über den Kontext des Benutzers sammeln und aggregieren. Von der Leistungsfähigkeit dieser Komponenten hängt also ab, ob die lernunterstützenden Dienste überhaupt agieren können. Deshalb muss diesem Teil der Architektur noch besondere Aufmerksamkeit gewidmet werden.

Als erster Schritt ist das Ziel dieses Kapitels die Zusammenstellung von Anforderungen aus den vorangegangenen Abschnitten an eine Kontextinfrastruktur insgesamt und an ein Verwaltungssystem für Kontextinformationen im Besonderen. Im vorangegangenen Teil wurden in die Betrachtung Kontextelemente wie Aufgabe, Projekt, Prozess und Aktivität, aber auch persönliche und soziale Parameter einbezogen, die üblicherweise nicht explizit abfragbar sind. Hierzu sind Heuristiken anzuwenden, die diese Informationen aus anderen beobachtbaren Anwendungen ermitteln. Indikativ wurden mögliche Quellen bereits in Abb. 8.5 aufgeführt. Dabei reicht das Spektrum von »einfachen« Kontextquellen, die bereits nahe am gewünschten Abstraktionsniveau sind, bis hin zu »spekulativen« Methoden, die beispielsweise den Stresszustand aus Dichte des Terminkalenders, Kommunikationsfrequenz etc. ermitteln oder versuchen, den Aufgabenkontext aus geöffneten Dokumenten zu erschließen.

Zusammenfassend ergeben sich hieraus die Hauptherausforderungen für eine Kontextverwaltung:



- Wie lässt sich **Qualität** sichern oder erreichen angesichts spekulativer und überlappender Ermittlungsmethoden?
- Wie lässt sich eine hinreichende **Vollständigkeit** erreichen angesichts von Ermittlungsmethoden, die nur unvollkommene Ergebnisse liefern? Was lässt sich durch die Einbeziehung von Hintergrundwissen oder Informationen aus vorangegangenen Zeiträumen ermitteln?

Daraus ergeben sich funktionale und nichtfunktionale Anforderungen im folgenden Abschnitt.

12.1. Anforderungen

12.1.1. Funktionale Anforderungen

KF1 **Anfragemöglichkeiten**. Die wichtige Funktion ist eine Möglichkeit, Kontextinformationen abzufragen. Hier lassen sich folgende Fälle unterscheiden:

- a) **Abrufen des aktuellen Kontextes**. Der Standard-Fall für eine kontextbewusste Anwendung ist der Abruf eines bestimmten Teiles des aktuellen Kontextes eines Benutzers, um diejenigen Werte zu ermitteln, die in den Adaptionprozess eingehen. Ein typisches Beispiel ist die Ermittlung der technischen Umgebung (Browser, Bildschirmgröße, Bandbreite, Verfügbarkeit von Lautsprechern), um die Anzeige von Lernobjekten anzupassen. Ein anderes Beispiel ist die Ermittlung des individuellen Stresszustandes und die vermutlich zur Verfügung stehende Zeit, um so die Auswahl von unterschiedlichen Präsentationsformen zu unterstützen.
- b) **Überprüfung von bestimmten Kontextinformationen**. Der Unterschied zum vorherigen Fall kommt durch die Unvollständigkeit zustande, wo Anfragen unterschiedliche Modalität haben können. Im vorangegangenen Fall werden vorhandene Fakten angefragt, während sich dieser Fall um Aussagen dreht, die von der Faktenlage gestützt werden. Eine typische Anfrage ist hier: »Ist es möglich, dass für den Benutzer X das Kontextmerkmal Y den Wert v hat?«. Dies kann sich auch auf historische Informationen beziehen (z.B. hat der Benutzer schon einmal in einer bestimmten Abteilung gearbeitet).
- c) **Abrufen von anderen Benutzern in einem bestimmten Kontext**. Gerade bei der Berücksichtigung der sozialen Dimension des Lernens ist es wichtig, den Kontakt zu Kollegen oder anderweitigen Kontakten herzustellen, die »in einem

ähnlichen Kontext« sich befinden (vgl. 10.1.4). Das System kann hier beispielsweise Personen empfehlen, die innerhalb der letzten vier Wochen dieselbe Aufgabe bearbeitet haben. Hierbei ist wichtig, dass dies sowohl benutzerübergreifend als auch die Historie umfassend geschieht.

- KF2 Aktionen durch Kontextänderungen auslösen.** Gerade für das proaktive Systemverhalten des Lernkoordinators ist es wichtig, eine zeitnahe Benachrichtigung über Änderungen von bestimmten Kontextinformationen zu erhalten. Ein Beispiel ist hier der Wechsel von Prozessaktivitäten oder Aufgaben, was ein idealer Zeitpunkt für die Initiierung von Lernprozessen darstellt.
- KF3 Speicherung über größere Zeiträume/Asynchronizität von Gewinnung und Nutzung.** Wie in der Einleitung ausgeführt kann bei Kontextinformationen auf höheren Abstraktionsebenen nicht vorausgesetzt werden, dass man durch unmittelbare Messungen bzw. darauf aufbauenden Berechnungen zu dem Zeitpunkt, zu dem bestimmte Kontextinformationen benötigt werden, diese auch ermitteln kann. Dies liegt im wesentlichen daran, dass man Kontextinformationen aus beobachteten Benutzeraktionen ableitet, diese Aktionen aber nicht notwendigerweise Kontextwechsel signalisieren, sondern nur Indizien für einen bestimmten Kontext liefern. Ein Beispiel kann die Kontextinformation »aktuelles Projekt« sein. Indizien hierfür können Kommunikationspartner, geöffnete Verzeichnisse o.ä. sein, wobei erst die Kombination dieser Indizien eine sinnvolle Aussage über den Kontext zulässt, so dass nicht ein kurzzeitige geöffnete Mail den Projektkontext verändern muss.
- KF4 Konfliktauflösung.** Durch die unterschiedlichen heuristischen Verfahren zur Kontextgewinnung ist von widersprüchlichen Kontextinformationen auszugehen. Allerdings würde der Umgang mit solchen Widersprüchen die Verfahren in Teil III erheblich komplexer gestalten, so dass sinnvollerweise eine Kontextinfrastruktur die Widersprüchlichkeit der Faktenlage verbirgt, indem sie nach außen eine konsistente Sicht erzeugt.
- KF5 Anreicherung.** Die Kontextinfrastruktur muss offen sein für Kontextanreicherungsverfahren, die die eingelieferte Information kombinieren und durch Hintergrundwissen weitere Fakten erschließen.
- KF6 Unterstützung von Ontologien.** Da wir uns für einen ontologiezentrierten Architekturstil entschieden haben, muss der Kontextverwaltungsdienst die Nutzung der Kontextontologie und der Domänenontologie allgemein erlauben.

12.1.2. Nicht-funktionale Anforderungen

- KNF1 **Qualitätszusicherungen/Umgang mit Imperfektion.** Die Kontextinfrastruktur muss trotz der heuristischen Gewinnungsmethoden Qualitätszusicherungen machen können. Dies betrifft zum einen die Aktualität von Daten. Ggf. unter Einbeziehung von Hintergrundwissen bzgl. der Veränderungsdynamik von Kontextmerkmalen muss sie dem Problem der veralteten Daten begegnen. Zum anderen muss sie mit der Unsicherheit der gelieferten Daten bewusst umgehen können.
- KNF2 **Entkopplung von Ermittlung, Anreicherung und Nutzung.** Die Architektur des Kontextdienstes muss eine klare Entkopplung der Ermittlungs-, Anreicherungs- und Nutzungsprozesse erlauben, um so das einfache Hinzufügen von Gewinnungs- und Anreicherungsheuristiken zu ermöglichen, ohne die nutzenden Anwendungen zu beeinträchtigen.
- KNF3 **Grundannahme einer offenen Welt.** Kontext ist immer ein unvollständiges Abbild der Situation, so dass Nichtwissen nicht gleichgesetzt werden darf mit Nichterfülltsein. Hier ist die »open world assumption« ein geeigneter Ansatz, bei dem es explizit positive und negative Fakten gibt. Ein triviales Beispiel ist hierbei das explizite Benutzer-Feedback (»ich bearbeite nicht diese Aufgabe«), doch gibt es auch weitere Verfahren, die negative Fakten generieren können, um so die Kontextqualität zu erhöhen. Auch für ein abgestimmtes Systemverhalten ist diese Differenzierung essentiell (z.B. um unnötige Störungen zu vermeiden), z.B. empfiehlt das System nur dann etwas proaktiv, wenn der Benutzer nach Faktenlage definitiv nicht in einer Präsentation ist, während bei Telefonaten man durchaus progressiver vorgehen kann und Empfehlungen dann anzeigt, wenn keine positiven Fakten für ein Telefonat vorliegen.
- KNF4 **Performanz und Skalierbarkeit.** Je nach Gewinnungsverfahren können schnell große Faktenmengen entstehen, auf die immer noch effizient zugegriffen werden muss. Dies gilt umso mehr für große Benutzerzahlen.

12.2. Stand der Technik: Kontextverwaltung

Für die Kontextinfrastruktur stehen zwei wesentliche Entwurfsentscheidungen an:

- Wie baut man eine Kontextverwaltung auf?
- Wie beschreibt man Kontextinformationen auf der Basis der Domänenontologie?
 - Was ist ein geeignetes Datenmodell (Schemaformalismus, Anfragemöglichkeiten)?

- Wie repräsentiert man in diesem Datenmodell Imperfektion?
- Wie repräsentiert man in diesem Datenmodell Historien?

Für diese Fragen soll zunächst der Stand der Technik aufbereitet werden im Hinblick auf die Fragen, inwieweit existierende Ansätze herangezogen werden können, um die Anforderungen zu erfüllen und wo die derzeitigen Lösungen noch Lücken haben. Am Ende des Kapitels wird der Handlungsbedarf herausgearbeitet, den die anschließenden Kapitel zu lösen versuchen.

12.2.1. Architektur von Kontextinfrastrukturen

Als erstes sollen existierende Architekturen von Kontextinfrastrukturen analysiert werden. Zusammenfassend lassen sich für die innere Architektur von Kontextinfrastrukturen drei verschiedene Klassen von Ansätzen identifizieren (vgl. [BP03]):

- **Widget-Ansatz.** Beim Widget-Ansatz werden Kontextquellen als sog. »widgets« (in Analog zu Steuerelementen an der Benutzerschnittstelle) gekapselt und dienen zur dezentralen Transformation, Filterung, Anreicherung und ggf. Speicherung der Historie. Anwendungen können sich aus den verfügbaren Widgets flexibel eine Auswahl zusammenstellen, bei denen sie sich anmelden. Damit kommt der Ansatz üblicherweise ohne eine zentrale Instanz aus; umgekehrt müssen allerdings Funktionen evtl. mehrfach für einzelne Widgets implementiert werden bzw. nicht alle Widgets verfügen über dieselbe Mächtigkeit an ihrer Schnittstelle.
- **Infrastrukturansatz.** Beim Infrastrukturansatz wird von einem klassischen Middlewareansatz ausgegangen. Neben die Kontextquellen treten wiederverwendbare Infrastrukturdienste, die sich um generische Kommunikations- und Fehlerbehandlungsfunktionalität kümmern. Damit wird das Hinzufügen neuer Kontextquellen deutlich einfacher und das Zugreifen auf sie konsistenter im Bezug auf die angebotene Dienstgüte.
- **Blackboard-Ansatz.** Der Blackboard-Ansatz ist für die Kontextverwaltung eine datenzentrische Weiterentwicklung des Infrastrukturansatzes. Hierbei wird explizit von einer zentralen datenspeichernden Komponente ausgegangen, auf die von Kontextquellen, weiterverarbeitenden Diensten und Anwendungen gleichermaßen zugegriffen werden kann. Die Komponenten kommunizieren und koordinieren sich über eine gemeinsame Datenhaltung. Dieser Ansatz wird von [Hec06b] verfolgt.

Widget-Ansatz

Der bekannteste Vertreter des Widget-Ansatzes ist das **Context Toolkit** von DEY [DSA01], der auf eine sog. Widget-Architektur aufbaut. Widgets abstrahieren Sensoren und sind für die Gewinnung von Kontextinformationen zuständig. Spezielle Aggregatoren und Integratoren, die man als Meta-Widgets verstehen kann, können diese für bestimmte Zwecke kombinieren (Aggregatoren) oder auf eine höhere Abstraktionsebene bringen (Integratoren) und entsprechen somit Mediatoren. Implizit ist ein virtueller Integrationsansatz unterstellt, so dass historische Kontextinformationen schwierig zu behandeln sind.

Infrastrukturansatz

Als Vertreter des Infrastrukturansatzes soll bei **CoBrA** (*Context Broker Architecture*) [CFJ04b] eine Middleware geschaffen werden, was mit der »Broker«-Metapher beschrieben und auf der Basis von Agententechnologie (FIPA) realisiert wird. Kontext soll zwischen unterschiedlichen Quellen, Systemen und Anwendungen austauschbar werden, wobei gleichzeitig Datenschutzanforderungen der Benutzer berücksichtigt werden. Basis für die Architektur ist eine Ontologie, die in der Ontologiesprache OWL beschrieben ist. Hier wird derzeit eine standardisierte Ontologie unter dem Namen SOUPA (Standard Ontology for Ubiquitous and Pervasive Applications) erarbeitet, die anwendungsunabhängige Konzepte und ihre Beziehungen definiert. Für Inferenzen wird ein ein zweischichtiges Vorgehen vorgeschlagen, das zum einen auf OWL-Inferenzmechanismen, zum anderen auf in PROLOG formulierten Regeln aufsetzt. Über Standard-Reasoning werden Inkonsistenzen erkannt. CoBrA ist ein vor allem aus Architekturperspektive interessanter Ansatz; allerdings fehlt z.B. eine Berücksichtigung der Imperfektion von Kontextinformationen (KNF1). Das bloße Erkennen von Inkonsistenzen in einer Ontologie wird der Komplexität der Problemstellung nicht gerecht.

Ähnliche Ansätze (vgl. auch [PLO⁺04]) sind z.B. Teil des Aura-Projekts [JS03] oder des PACE-Projektes [HIM05] (die beide auf adaptive Infrastrukturen für das Pervasive Computing abzielt) oder des AMIGO-Projektes [MEJ⁺06] (im Bereich der Heimautomatisierung).

In Abschnitt 9.1.1 wurde bereits der Architekturentwurf von HENRICKSEN UND INDULSKA [HI06] vorgestellt, der in eine ähnliche Richtung geht und auf der Ebene der Kontextinfrastruktur vier Ebenen differenziert: (1) Kontextmessung, (2) Kontextaufnahme, (3) Kontextmanagement und (4) Anfrage.

Blackboard-Ansatz

Im Gegensatz zu den bisher betrachteten Ansätzen geht **MADAM** [Sat01] von einem zentralisierten *Context Repository* aus, das Kontextgewinnung und -nutzung nach dem ma-

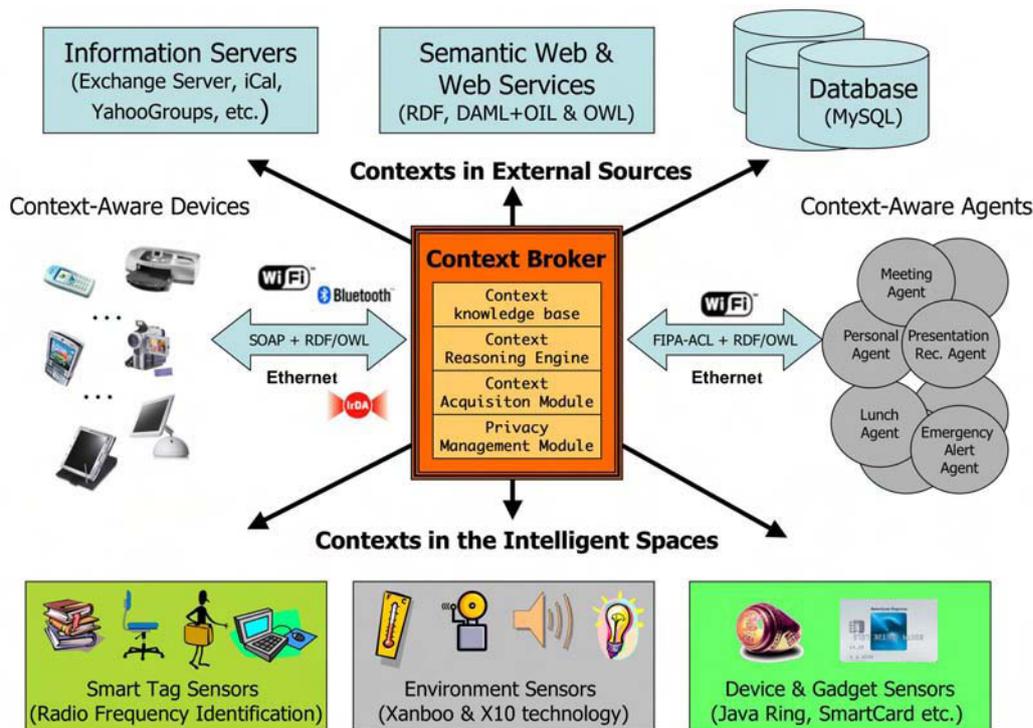


Figure 1: An intelligent context broker acquires context information from devices, agents and sensors in its environment and fuses it into a unified context. **Abbildung 12.1: COB-Architektur (Theobald)**

of the acquired contextual information, and the meta-data for describing the storage structure of the terialisierten Ansatz an. Dieser Kontextspeicher ermöglicht auch die asynchrone Benachrichtigung über Kontextänderungen.

2. *Context-reasoning engine.* A logical inference engine for reasoning over the acquired contextual information. Ein weitergehender Ansatz ist der von HECKMANN ([Hec06b], [Hec06a], vgl. Abb. 12.2) aus dem Bereich der Benutzermodellierung. Er zielt auf einen Benutzermodellierungsservice ab, der in der Lage ist, Benutzerinformationen zwischen unterschiedlichen Anwendungen auszutauschen. Der Fokus liegt daher (1) auf der verteilten Speicherung, (2) auf der Integration über eine gemeinsame Ontologie (GUMO, [HSB⁺05]) und (3) auf der Wahrung von Datenschutz. Es werden folgende Dienste angeboten: ein *Situation Server*, der den zentralen Zugriff auf die Benutzerinformationen bereitstellt, ein *Situation Adder*, der eintreffende Informationen aggregiert, ein *Weblog Filter*, der die relevanten Ereignisse für das Einsammeln von Benutzerinformationen kümmert, einer Komponente für *Conflict resolution* from the high-level functional components. This middle-ware system design is aimed to improve the *Inference Engine*, der die Benutzerinformationen anreichern kann. Zusätzlich existiert ein *Ontologie-Reasoner*.

4. *Privacy-management module.* This component manages the users' privacy policies and controls the sharing of their private information. It is responsible for enforcing the appropriate privacy policies when the context broker attempts to share certain user information. When sharing information, if

new statements. The latter is described in the subsequent subsection 8.1.3, directly after the service architecture is looked at from its procedural view.

8.1.2 Procedural View to the User Model Service

Figure 8.2 shows the input and output information flows *Add*, *Request* and *Report* of the USERMODEL SERVICE. They are designed as yellow arrows with a short description of their format. The numbers in the orange ovals present the procedural order. Number (1) visualizes

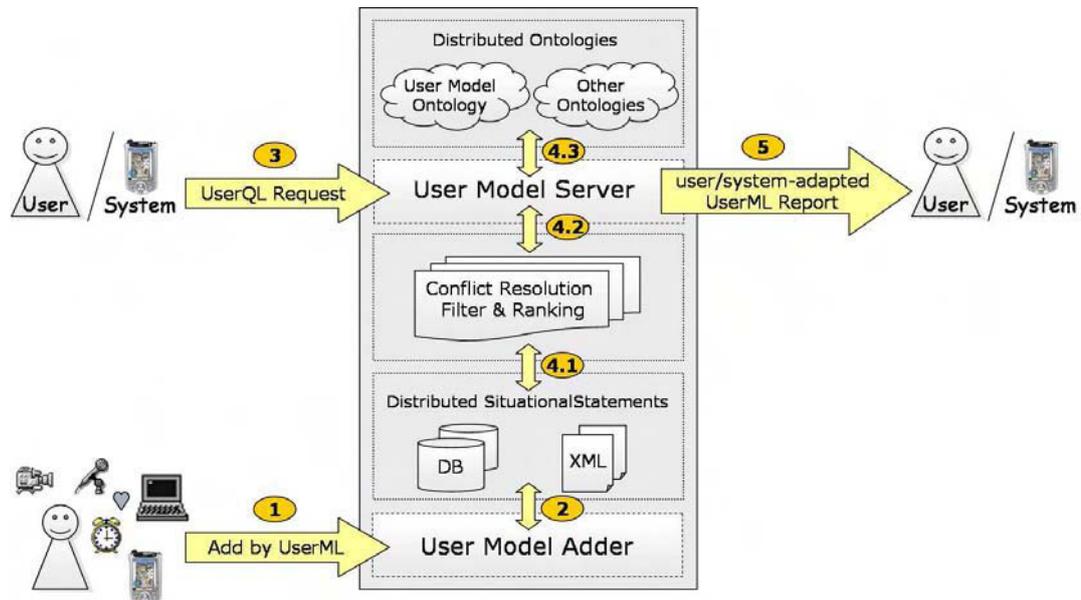


Figure 8.2: Procedural view to the USERMODEL SERVICE

Abbildung 12.2.: Architektur des Ansatzes von Heckmann [Hec06b]

12.2.2. Datenmodelle für Kontextinformationen

Bei der Analyse der benutzten Datenmodellen für die Repräsentation von und den Umgang mit Umgang mit Kontextinformationen ist ein klarer Trend in Richtung ausdrucks-mächtigerer Formalismen erkennbar ([SLP04], [Kle02], [LB04], [Cha02], [Str03]). Während den ersten Kontextmodellen Schlüssel-Wert-Paare ausreichten, wurde frühzeitig auf XML-basierte Sprachen wie ContextML [Rya99] gesetzt, um auch komplexer strukturierte Kontextinformationen repräsentieren zu können, z.B. Entitäten mit Eigenschaften. In den neueren Ansätzen dominieren ontologiebasierte Ansätze, die sich allerdings hinsichtlich der Ausdrucksmächtigkeit des verwendeten Ontologieformalismus stark unterscheiden:

- Der bereits erwähnte Ansatz von HECKMANN ([Hec03a], [Hec03b], [Hec06a]) konzentriert sich auf die Erarbeitung einer universellen Benutzermodellierungsontologie (GUMO - General User Modeling Ontology) auf der Basis von RDF(S) (sog. »situational statements«, [Hec03a]), bei dem die Tripelstruktur um zusätzliche Meta-informationen ergänzt wird. Zu diesen Metainformationen gehören: temporale Einschränkungen (bzgl. der Gültigkeit), räumliche Einschränkungen, Datenschutzeinstellungen, eine Liste von Indizien, aus denen die Aussage geschlossen wurde, sowie ein Konfidenzfaktor.

- STRANG [SLPF03] setzt auf ein eigenes Modell (*Aspect-Scale-Context*), das sich besonders um die Abbildbarkeit unterschiedlichen »Skalen« (also Wertebereiche) konzentriert. Dieses Modell wird zur Ausführung in DAML+OIL bzw. OWL-DL übersetzt.
- POWER [Pow03] setzt auf TopicMaps [ISO02].
- Die »Standard Ontology for Ubiquitous and Pervasive Applications« (SOUPA) [CFJ04a] setzt auf OWL-DL, ähnlich wie andere Ansätze auf OWL-DL wie z.B. [GWPZ04], [WGZP04].

Der Ansatz von HECKMANN erfüllt viele der Anforderungen und kann als Ausgangspunkt für die weitere Entwicklung dienen; allerdings ist für KNF1 das Instrument vordefinierter Gültigkeitszeiträume nicht ausreichend, um dem Veralten von Daten zu begegnen: Gerade bei wechselnden Aufgaben ist nicht zu prognostizieren, wie lange dies zutrifft. Weiterhin fehlt eine ausreichende Berücksichtigung der Folgen der *open world assumption*, die als Grundlage für den verteilten Ansatz erwähnt wird; denn der Ansatz erlaubt keine negativen Fakten. Da der Ansatz außerdem stark auf (hauptspeichergestütztes) Ontologie-Reasoning aufsetzt, ist bei größeren Datenmengen mit Problemen zu rechnen; Tests mit 40.000 Statements reichen hier nicht aus; diese Anzahl ist bereits bei Nutzern bei einem angenommenen 8-Stunden-Arbeitstag innerhalb einer Arbeitswoche erreicht, wenn man z.B. von einer Aktualisierung pro Minute und Benutzer ausgeht. Zudem fehlt die Möglichkeit für asynchrone Benachrichtigungen.

Repräsentation von Imperfektion und Qualitätsmerkmale

Obwohl wie schon erwähnt die Unvollkommenheit und temporale Abhängigkeit als Problem durchaus wahrgenommen wird, werden in den Ansätzen in der Literatur Qualitätskriterien oder die Beschreibung der Imperfektion von Kontextinformation überraschend stark vernachlässigt. Im wesentlichen lassen sich drei Ansätze identifizieren:

- BUCHHOLZ ET AL [BKS03] untersuchen Qualitätskriterien für Kontextinformationen in Ergänzung zu Dienstgüte-Betrachtungen. Als Kriterien definieren sie: Genauigkeit, Korrektheitswahrscheinlichkeit (bezogen auf Kontextinformationen), Vertrauenswürdigkeit (bezogen auf Kontextquellen), Granularität und Aktualität.
- JUDD UND STEENKISTE [JS03] führen sogenannte Meta-Attribute ein, mit denen Anwendungen, die Kontextinformationen anfordern, ihre Qualitätskriterien spezifizieren können. Zu diesen Meta-Attributen gehören Genauigkeit, Gewissheit, letzte Aktualisierung und Aktualisierungsrate.
- Erst HENRICKSEN [HI04] hat sich systematischer mit der inhärenten Imperfektion beschäftigt und folgende Formen der Imperfektion identifiziert: unbekannte Werte,

widersprüchliche Werte, ungenaue Werte und fehlerhafte Werte. Weiterhin werden Kontextinformationen nach ihrer Quelle und der Persistenz klassifiziert in gemessen (*sensed*), statisch, gesammelt (*profiled*) und abgeleitet und entlang dieser Klassifikation die Ursachen für Imperfektion analysiert. Um mit der Imperfektion umzugehen, wird ein auf »Object Role Modelling (ORM)« basiertes Kontextmodell vorgestellt, das um Qualitätsparameter ergänzt wird. Fehlende Werte werden über Null-Werte repräsentiert und über die »closed-world assumption« verarbeitet.

Allerdings lassen sich in der Literatur zum allgemeinen (also nicht auf »Kontext« als Anwendungsfall beschränkten) Umgang mit imperfekten Daten einige Ansätze identifizieren. Diese lassen sich nach unterschiedlichen Aspekten der Imperfektion auftrennen. Klassisch ist hier die Trennung in Ungewissheit/Unsicherheit (*uncertainty*, die Gültigkeit in der Realwelt von Fakten in unserem Modell ist nicht sicher), Unvollständigkeit (*incompleteness*, wir besitzen nicht alle Fakten aus der Realwelt, die in unser Modell abbildbar sind) und Ungenauigkeit (*imprecision*, (statt eines exakten Wertes in einem bestimmten Faktum haben wir nur eine gröbere Granularität, z.B. ein Intervall statt eines Wertes), wie sie z.B. von [BT85] (ähnlich [Mot94], wenn dieser auch Unsicherheit in zwei unterschiedlichen Bedeutungen verwendet) vorgenommen wird. [BP93] fügt hier Widersprüchlichkeit (*inconsistency*) hinzu. [Ber02] fügt hier noch eine ganze Reihe anderer Arten der Imperfektion hinzu, wobei in der Klassifikation nicht klar ist, ob hier nach Ursachen oder anderen Kriterien klassifiziert wird.

Für den Umgang mit Imperfektion gibt es zahlreiche Ansätze, die z.B. [PH98] entlang der Arten von Imperfektion gruppiert. Für unsere Zwecke sind vor allem Unsicherheit (KNF1), Unvollständigkeit (KNF3) und Widersprüchlichkeit (KF4) von Interesse, so dass wir diese im folgenden genauer untersuchen.

Unsicherheit

Für den Umgang mit **Unsicherheit** gibt es probabilistische und evidenzbasierte Ansätze.

Probabilistische Ansätze gehen statt eines exakten Wertes von einer Wahrscheinlichkeitsverteilung auf einer bestimmten Grundmenge aus. Im Bereich der probabilistischen Datenbanken gibt es dabei zwei Grundansätze: probabilistische Relationen und probabilistische Attribute. Bei probabilistischen Relationen ([CP87], [FR97], [DS98], [DRS01]) wird pro Primärschlüsselausprägung eine Wahrscheinlichkeitsverteilung auf möglichen Tupeln zugelassen. Bei probabilistischen Attributen ([BGMP92], [DS96]) werden bestimmte Attribute als probabilistisch deklariert, so dass ihnen statt eines exakten Wertes eine Wahrscheinlichkeitsverteilung (z.B. durch eine Dichtefunktion) zugewiesen werden kann, was sich eher für die Repräsentation von Ungenauigkeit als zur Repräsentation von Unsicherheit eignet.

Im Bereich der deduktiven Datenbanken (z.B. probabilistisches Datalog, [Fuh95]) wird jedem Faktum und jeder Regel eine Wahrscheinlichkeit zugewiesen; unter Zuhilfenahme der Annahme der statistischen Unabhängigkeit von Ereignissen werden dann die Regeln ausgeführt. Eine Erweiterung dieses Ansatzes ([FR97]) basiert auf einer vierwertigen Logik, die den Wechsel von einer geschlossenen Welt hin zu einer offenen Welt ermöglicht und hierzu auch negative Fakten einführt, um auch den Aspekt der Unvollständigkeit abbilden zu können. Dieser Ansatz wurde in [NF04] dazu benutzt, um eine probabilistische Erweiterung von DAML+OIL, einem Vorläufer der Ontologiesprache OWL, zu definieren. Ähnliche Ansätze existieren auf der Ebene von RDF von [USM06] und [Fuk05].

Evidenzbasierte Ansätze, deren prominentester Repräsentant die Dempster-Shafer-Theorie ist, sind Erweiterungen der Wahrscheinlichkeitstheorie. So kann Dempster-Shafer zwischen Unsicherheit und Unwissen differenzieren (vgl. z.B. [RN03], [SF02]). Die Grundidee im Gegensatz zur Wahrscheinlichkeitstheorie ist hierbei, dass sie statt die Wahrscheinlichkeit einer Aussage zu berechnen, die Wahrscheinlichkeit ermitteln, dass die Beweise die Aussage stützen. Sie wird besonders für die Zusammenführung von ungewissen und widersprüchlichen Daten aus unterschiedlichen Quellen benutzt [RL02]. Ihr Ergebnis ist für eine bestimmte Wertemenge A ein Intervall $[Pl(A), Bel(A)]$, wobei $Pl(A)$ die Möglichkeit angibt, dass die Beweislage A zulässt, und $Bel(A)$ angibt, inwieweit die Beweislage A stützt [Wu03]. Damit eignet sich die Theorie besonders für die Kombination von unsicheren Daten aus unterschiedlichen, potentiell widersprüchlichen Quellen. Allerdings ist i.d.R. mit exponentieller Berechnungskomplexität zu rechnen.

Unvollständigkeit

Klassische Datenbankansätze arbeiten nach der *closed world assumption*, bei denen grundsätzlich davon ausgegangen wird, dass alle zutreffenden Fakten auch in der Datenbasis enthalten sind. Demgegenüber wird in Semantic-Web-Ansätzen stärker die offene Welt betont, so dass aus dem Nichtvorhandensein eines Faktums nicht geschlossen werden kann, dass dieses nicht zutrifft. In klassischen Datenbankansätzen besteht daher keine Notwendigkeit, das Nichtzutreffen von Fakten zu repräsentieren – in diesem Fall fehlt einfach das entsprechende Tupel in einer Relation. Demgegenüber sollten in Ontologiesprachen diese zur Verfügung stehen, um sicheres Wissen über das Nichtzutreffen zu repräsentieren. Allerdings ist dies weder in RDFS, noch in der OWL-Familie vorzufinden. Erst im derzeit in der Diskussion befindlichen OWL 2.0 finden sich *negative object property assertions*.

Temporale Aspekte

Für den Umgang mit temporalen Aspekten existieren zwei unterschiedliche Grundansätze. Bei **temporalen Datenbanken** geht man davon aus, dass sich die Daten über die Zeit hin-

weg ändern und man nicht nur den jeweils aktuellen Zustand, sondern auch vergangene und ggf. auch zukünftige Zustände zugreifbar halten will, um so z.B. die Auswertung der Historie zu ermöglichen. Dies bedeutet, dass man den Zeitbezug der Daten als Bestandteil des Datenmodells betrachtet wird. Die meisten Ansätze gehen von zwei Arten des Zeitbezuges aus ([JCGS92])¹: (1) Transaktionszeitpunkt (Zeitpunkt, zu dem die Änderung ins Datenbanksystem eingebracht wurde) und (2) Gültigkeitszeitraum (Zeitraum, in dem das Datum in der Realwelt gültig ist), was dann oft auch als *bitemporal* bezeichnet wird.

Demgegenüber steht der Ansatz des temporalen Schließens (*temporal reasoning*), bei dem Zeitangaben und (relative) Beziehungen zwischen ihnen für das logische Schlussfolgern benutzt werden (vgl. z.B. [AF00], [AFWZ02]), z.B. im Bezug auf die Reihenfolge von Ereignissen und die gleichzeitige Verwendung von absoluten und relativen Zeitangaben.

Für das Kontextmanagement spielt das temporale Schließen keine erkennbare Rolle, so dass man sich auf Ansätze im Bereich der temporalen Datenbanken konzentrieren kann.

12.3. Lösungsansatz und Architektur

Wie der vorangegangene Abschnitt gezeigt hat, bestimmen architekturelle Aspekte sehr stark die Lösungen für den Umgang mit Kontextinformationen, so dass wir als ersten Schritt auch einen groben Architekturentwurf erarbeiten wollen. Dabei steht das Grundproblem der Trennung der unterschiedlichen Funktionen im Vordergrund, die in den verschiedenen Ansätzen unterschiedlich aufgeteilt sind.

12.3.1. Grundprobleme der Architektur

Grundlage für die Architektur der Kontextverwaltung ist die Forderung von KNF2 nach einer klaren Trennung der drei Aufgaben der Kontextverwaltung:

- **Sammlung und Speicherung von Kontextinformationen (KF3).** Dies umfasst die Anbindung von unterschiedlichen Kontextquellen und die Speicherung ihrer Daten über die Zeit hinweg.
- **Anfragemöglichkeiten für Kontextinformationen (KF1, KF2, KF4).** Die Kontextverwaltung muss nutzenden Anwendungen adäquate Schnittstellen für den Zugriff bereitstellen. Adäquatheit bezieht sich dabei sowohl auf die Mächtigkeit, als auch auf Qualitätsaspekte.
- **Anreicherung von Kontextinformationen (KF5).** Dies wird oft auch als *context reasoning* bezeichnet und umfasst das Schließen/Berechnen von Kontextinformationen

¹vgl. auch *The Consensus Glossary of Temporal Database Concepts*, <http://www.cs.aau.dk/~csj/Glossary/index.html>

auf höherer Abstraktionsebene sowie die Vervollständigung mit Hilfe von Hintergrundwissen.

Diese Trennung, so einfach und einsichtig sie auf den ersten Blick sein mag, wirft allerdings Probleme auf:

- **Sammlung/Speicherung vs. Anreicherung.** Bei Ansätzen, die davon ausgehen, dass Kontextinformationen nur von einer niedrigen Abstraktionsebene auf eine höhere gebracht werden müssen und es dann ausreicht, *high-level* Kontextinformationen zu speichern, steckt die (oft implizite) Annahme dahinter, dass a priori bekannt ist, was *low-level* und was *high-level* ist: *low-level* wird von Kontextquellen geliefert, und *high-level* von Anwendungen genutzt. Diese klare Trennung macht zwar im Fall von Sensoransätzen (vgl. Abb. 12.3) Sinn, ist im vorliegenden Fall nicht sinnvoll, da sie unnötige Einschränkungen bzgl. des Verhältnisses von Gewinnung und Nutzung macht.
- **Anfrage vs. Datensammlung/Speicherung.** Zusätzlich zu dem bereits im vorangehenden Abschnitt erwähnten Problem der Asynchronizität von Gewinnung und Nutzung ist auch eine teilweise Vorwegnahme von Anfrageaspekten problematisch, wie sie z.B. durch die Auflösung von Widersprüchen direkt beim Einbringen in die Datenbasis geschehen kann. Dies hängt damit zusammen, dass die Konfliktauflösung heuristischer Natur ist (und somit flexibel konfigurierbar bleiben muss) und sich bei neu eintreffenden Fakten nicht-monoton verhalten kann.
- **Anfrage vs. Anreicherung.** Während besonders ontologiegestützte Reasoning-Ansätze darauf setzen (vgl. Abb. 12.3), dass man sowohl Anfragen als auch die Anreicherung durch Reasoning lösen kann, ist das Problem dieser Kopplung, dass man sich damit auf einen einzigen Formalismus festlegt, während unterschiedliche Ansätze für unterschiedliche Teilprobleme sinnvoll sein können. So kann z.B. eine statistische Analyse der Historie (z.B. für die Ermittlung bestimmter Muster und darauf aufbauender Vorhersage) kaum sinnvoll als Reasoning-Problem formuliert und gelöst werden. Auf der anderen Seite bieten deskriptive Regelsprachen Vorzüge für die Wartung vieler, einfach strukturierter Anreicherungsbeziehungen.

12.3.2. Überblick über die Architektur

Blackboard-Architektur zur Entkopplung von Gewinnung, Speicherung und Anreicherung

Für die Entkopplung von Sammlung/Speicherung und Anreicherung bietet sich die Blackboard-Architektur [HR85] an, die ein Architekturmuster für die kollaborative Problemlösung darstellt und aus folgenden Rollen besteht:

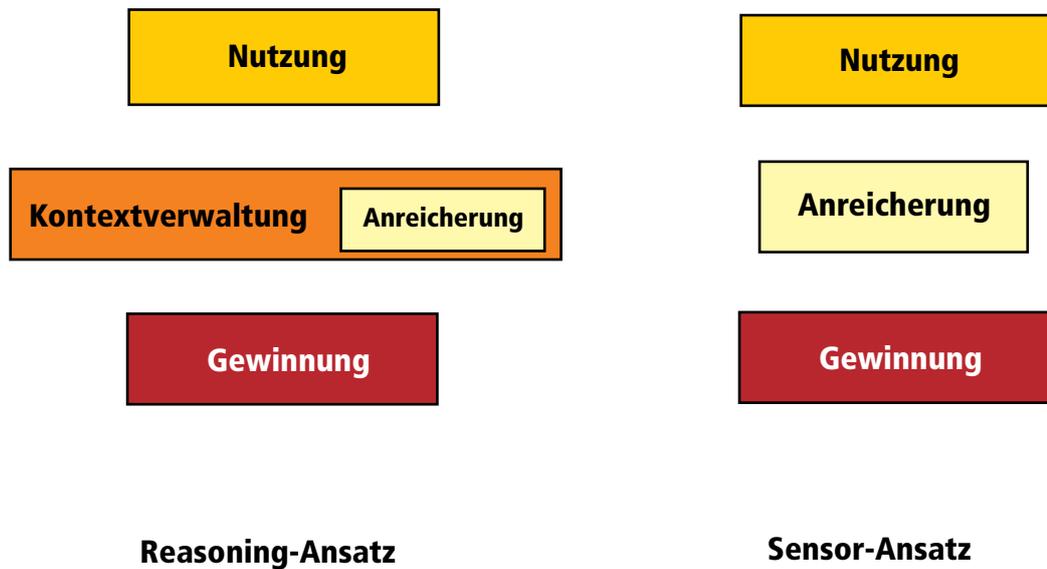


Abbildung 12.3.: Architektur der Kontextverwaltung: Alternativen

- Ein Blackboard (Tafel), auf das partielle Ergebnisse während des Problemlösungsprozesses geschrieben werden
- Wissensquellen, die spezifisches Expertwissen einbringen, indem sie auf der Basis der Informationen auf dem Blackboard und ihrem Expertenwissen ein Teilproblem zu lösen versuchen und ihr Ergebnis auf das Blackboard schreiben.
- Eine Kontrolleinheit, die die Interaktion zwischen Wissensquellen und Blackboard steuern.

Übertragen auf das Problem des Kontextmanagement lässt sich das »Blackboard« als eine Datenbank von Kontextfakten interpretieren. Die »Wissensquellen« kapseln Wissen und Strategien, um auf der Basis der vorhandenen Kontextfakten neue Fakten zu erschließen (was oft als »context reasoning« bezeichnet wird), den Kontext also anzureichern. Aus diesem Grund werden im folgenden diese Komponenten als »Anreicherungsagenten« bezeichnen.

Wie sieht nun die Steuerung der Anreicherungsagenten aus? Sie hat als wesentliche Aufgaben (a) zu verhindern, dass durch das unkoordinierte Zusammenwirken Inkonsistenzen entstehen, und (b) die Zielorientierung zu fördern. Da die Verhinderung von Inkonsistenzen kein wesentliches Problem darstellt (sie entstehen ohnehin durch die unterschiedlichen Gewinnungsverfahren), ist nur die Frage zu klären, wann welcher Agent zur Problemlösung beitragen kann. Dies lässt sich am einfachsten durch eine Kombination mit

dem Publish-Subscribe-Muster [GHJV96] erreichen, bei der die Kontrolleinheit das »Subjekt« darstellt, bei dem sich die Anreicherungsagenten als »Beobachter« registrieren mit der Zusatzinformation, zu welchen Änderungsereignissen sie benachrichtigt werden wollen.

Damit ist eine »Formalismusneutralität« für die Kontextanreicherung erreicht, die gleichzeitig auch die Entkopplung von Anfrage und Anreicherung vollzieht, d.h. die Architektur erlaubt die Verwendung von unterschiedlichen Ansätzen zur Anreicherung, was von statistischen über logische bis hin zu prozedural formulierten Ansätzen reicht.

Zwei Schichten der Kontextverwaltung für die Entkopplung von Nutzung und Speicherung/Anreicherung

Der Entkopplung von Anfrage und Sammlung/Speicherung ist dadurch beizukommen, dass man die Einbringung in eine Datenbasis von der Herstellung einer konsistenten Sicht trennt: Man führt eine Schicht von Kontextfakten ein, in die *alle* Fakten, die von Kontextquellen und Anreicherungsagenten geliefert werden, eingebracht werden. Darauf aufbauend stellt die Kontextinformationsschicht durch Auflösung auftretender Konflikte eine konsistente Sicht her. Anreicherungsagenten können es sich dabei aussuchen, ob sie durch Änderungen auf der Faktenebene oder auf der Informationsebene aktiviert werden wollen. Damit sind auch komplexe Auswertungen (z.B. statistische Verfahren) möglich, die nicht auf eine konsistente Sicht angewiesen sind.

Resultierende Architektur

Aus diesen Überlegungen ergibt sich damit die Architektur der Kontextverwaltung (vgl. Abb. 12.4), die aus folgenden Bausteinen besteht:

- **Kontextquellen** liefern Kontextfakten, entweder auf Anfrage (*Pull-Quellen*) oder von sich aus (*Push-Quellen*).
- Der **Kontextfaktendienst** hat zur Aufgabe, *Kontextfakten*, wie sie von den Kontextquellen eingeliefert oder abgeholt werden, zu speichern. Neben einfachen Zugriffsmöglichkeiten auf die Datenbasis ist er auch in der Lage, andere Komponenten über bestimmte Änderungen zu benachrichtigen.
- Beim Kontextfaktendienst registrieren sich **Anreicherungsagenten** für diejenigen Änderungen, auf die sie reagieren können. Sie berechnen dann zusätzliche Kontextinformationen (z.B. durch Abstraktion, aber auch negative Fakten als Korrektur) und fügen das Ergebnis dem Kontextfaktendienst hinzu. Hier können sehr unterschiedliche Verfahren eingesetzt werden, neben deklarativen Regelformalismen auch Bayessche Netze, Hidden-Markov-Modelle oder auch Mining-Verfahren.

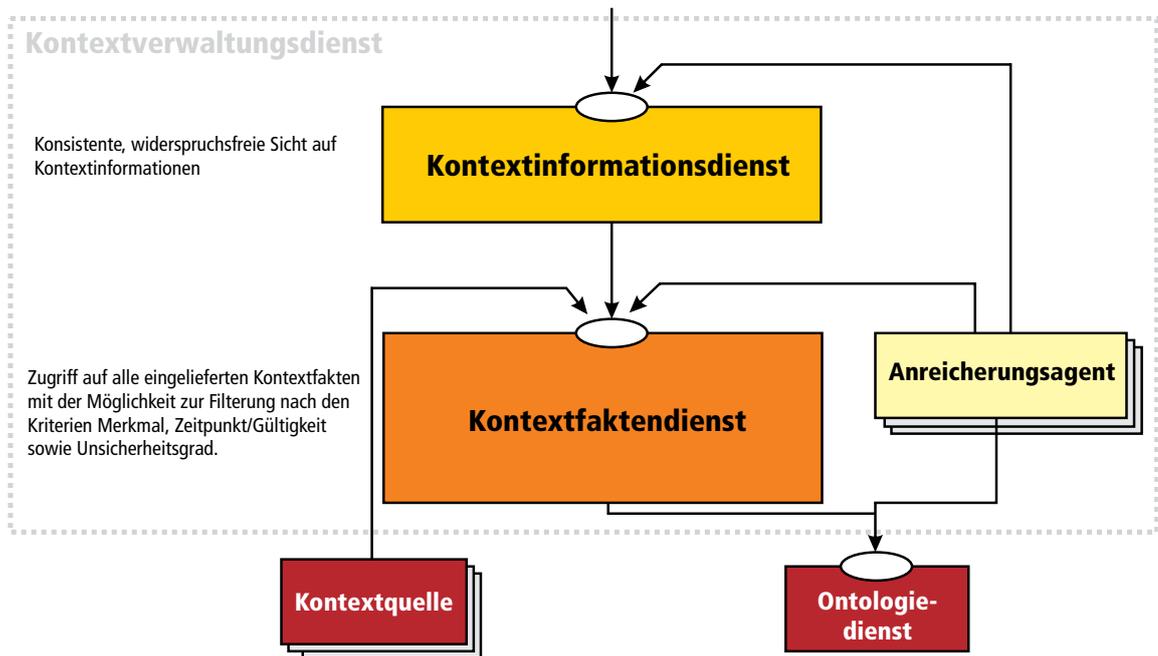


Abbildung 12.4.: Kontextverwaltungsinfrastrukturdienst: Konzeptionelle Architektur

- Der **Kontextinformationdienst** nutzt den Kontextfaktendienst und erbringt Anfragebearbeitungsdienste für Nutzer der Kontextverwaltung. Hierzu bereinigt er im Rahmen der Anfragebearbeitung die Faktenmenge durch Konfliktauflösung.
- Für die Nutzung von in Ontologien repräsentiertem Hintergrundwissen können sowohl Anreicherungsagenten als auch der Kontextfaktendienst auf den **Ontologiedienst** zurückgreifen, der ihnen Schlussfolgerungen durchführt und ihnen die Domänenontologie zugreifbar macht.

12.4. Fazit

Mit der erarbeiteten Architektur ist die Grundlage für einen flexiblen Kontextdienst gelegt. Besonderer Fokus lag dabei auf der Entkopplung unterschiedlicher Aspekte des Umgangs mit Kontextinformationen: Gewinnung, Speicherung, Anreicherung und Nutzung. In den folgenden beiden Kapitel wird nun gezeigt, wie diese Architektur es ermöglicht, mit Unsicherheit, den damit verbundenen potentiellen Widersprüchen und der temporalen Perspektive zurechtzukommen. Hierzu ist ein entsprechendes Datenmodell zu entwickeln mit den entsprechenden Anfragemöglichkeiten.

13.

Kontextfaktenebene

In diesem Kapitel geht es um die Kontextfaktenebene in der Architektur des Kontextdienstes. Sie hat die Funktion, Kontextfakten, wie sie von Kontextquellen eingeliefert werden, zu sammeln und per Anfragen, die sich auf temporale Aspekte oder ihre Unsicherheit beziehen, effizient bereitzustellen.

Die Schnittstelle des Dienstes ist dabei relativ einfach:

Kontextfaktendienst
<pre>List<ContextFact> getFacts(FactQuery) void addContextFact(ContextFact)</pre>

Der Dienst bietet also nur Filteroperationen auf gesammelten Fakten an und überprüft nicht auf Widerspruchsfreiheit von Aussagen auf der Basis der Faktenlage, so dass Dienstanutzer positive und negative Fakten gleichzeitig als Ergebnis erhalten können.

- Wie lassen sich Kontextfakten adäquat repräsentieren?
- Wie sehen angemessene Anfragen aus?
- Wie geht man mit Unsicherheit und temporalen Aspekten um?

Im folgenden ist zu konkretisieren, wie ein angemessener Repräsentationsformalismus für Kontextfakten (*ContextFact*) und eine angemessene Anfragesprache auszusehen kann. Danach ist zu klären, wie eine effiziente Implementierung aussehen könnte.

13.1. Datenmodell für Kontextfakten

Wie im vorangegangenen Kapitel herausgearbeitet, erscheint RDF als Ausgangspunkt für das Datenmodell, das auf der Kontextfaktenebene zum Einsatz kommt, erfolgversprechend. Allerdings reicht die einfache Tripel-Struktur nicht aus, und sie muss ergänzt werden um temporale und Unsicherheitsperspektive.

Beginnen wir auf Schemaebene bei den Kontextmerkmalen. Sie entsprechen im wesentlichen den *Properties* aus RDF(S). Sie besitzen einen eindeutigen Identifikator und einen Wertebereich. Hier unterscheiden wir nicht zwischen Literalen und Ressourcen und sehen auf dem jeweiligen Wertebereich Prädikate vor, die sich im Rahmen von Anfragen auswerten lassen. Formaler lässt sich das wie folgt fassen:

Definition 13.1 (Datentyp) Ein *Datentyp* V ist ein Tupel (S, P) , wobei S eine nichtleere Menge von möglichen Werten und P eine Menge von ein- oder mehrstelligen Prädikaten, die sich auf Werte aus S anwenden lassen.

Definition 13.2 (Kontextmerkmal) Ein *Kontextmerkmal* f ist ein Tupel $f = (uri, V)$, wobei uri ein eindeutiger Identifikator, V ein Datentyp ist. Im folgenden bezeichnet $value-space(f) = value-space(uri, V) = S$ den Wertebereich eines Kontextmerkmals.

Analog zur Möglichkeit von RDFS, Property-Hierarchies zu definieren, kann auf die Menge der Kontextmerkmale eine **Merkmalsstruktur** aufgeprägt werden, um semantische Verfeinerungen auszudrücken, wie sie in Kapitel 8 eingeführt wurden. Diese können dazu genutzt werden, dass für Anfragen von Anwendungen auf allgemeinere Merkmale auch die für speziellere Merkmale vorliegenden Werte geliefert werden können.

Definition 13.3 (Merkmalsstruktur) Eine *Merkmalsstruktur* ist eine azyklische Relation $H \subseteq (F \times F)$ auf der Menge der Kontextmerkmale F . H^* bezeichnet die reflexive und transitive Hülle von H .

Für zwei Kontextmerkmale f_1 und f_2 aus F gilt:

$$f_1 \triangleright f_2 := (f_1, f_2) \in H^*$$

was bedeutet, dass f_1 ein übergeordnetes Merkmal von f_2 darstellt.

Die Semantik der Merkmalsstruktur ist, dass das untergeordnete Kontextmerkmal das übergeordnete semantisch verfeinert. Es soll also gelten, dass aus dem Wert des untergeordneten auf den Wert des übergeordneten geschlossen werden. Dies wird im Rahmen der Definition der Semantik der Anfragen weiter unten noch weiter präzisiert. An dieser Stelle ist nur wichtig, dass die Merkmalsstruktur verträglich sein muss mit den Wertebereichsdefinitionen der Merkmale:

Definition 13.4 (Verträglichkeit der Merkmalsstruktur) Eine Merkmalsstruktur H ist *verträglich* mit einer Menge von Kontextmerkmalen F , wenn gilt:

$$\forall (f_1, f_2) \in H : \text{value-space}(f_2) \subseteq \text{value-space}(f_1)$$

Damit sind alle Bausteine für die Definition eines Faktenschemas eingeführt:

Definition 13.5 (Kontextfaktenschema) Ein *Kontextschema* C ist ein Tupel $C = (F, H)$, wobei F eine Menge von Kontextmerkmalen und H eine mit F verträgliche Merkmalsstruktur ist.

Was jetzt noch fehlt ist die Struktur der Instanzdaten. Hier muss das simple Tripelmodell um (1) Zeit- und (2) Unsicherheitsaspekte sowie der Möglichkeit, (3) negative Fakten zu repräsentieren, erweitert werden.

Im Rahmen von temporalen Erweiterungen für RDF [GHV05] wurde zwischen dem einfachen Versionierungsansatz und dem Zeitstempelansatz unterschieden. Bei der Versionierung wird bei jeder Änderung eine neue Version angelegt, was bei den zu erwartenden häufigen Änderungen unpraktikabel ist. Das Zeitstempelmodell, das Tripel jeweils mit Zeitstempeln versieht, ist hier flexibler. Hiermit ist allerdings nicht ohne weiteres auszudrücken, dass ein Faktum nur für einen bekannten Zeitraum seine Gültigkeit hat. Hier lohnt ein Rückgriff auf Konzepte aus temporalen Datenbanksystemen. Hier wird (vgl. [JCGS92]) u.a. zwischen zwei Zeitaspekten unterschieden: dem Transaktionszeitpunkt (wann das Tupel in die Datenbasis eingebracht wird) und dem Gültigkeitszeitraum in der Realwelt, die wir auch für unser Modell übernehmen.

Für die Repräsentation von Unsicherheit kann auf das bekannte Konzept der Konfidenz/Korrektheitswahrscheinlichkeit zurückgegriffen, wie es auch von HECKMANN in [Hec06b] verwendet wird. Hierbei wird die Aussage annotiert mit der Korrektheitswahrscheinlichkeit zum Transaktionszeitpunkt.

Schließlich fehlt noch die Möglichkeit, auch negative Fakten einzubringen. Der einfachste Fall hierbei ist explizites Nutzerfeedback, wo ein Benutzer angibt, dass bestimmte Aussagen über seinen aktuellen Kontext falsch sind. MORBIDONI hat existierende Ansätze zur negativen Aussagen in RDF und OWL analysiert [MPT07] und kommt zu dem Schluss, dass alle nicht besonders praktikabel sind, weil sie umständlich oder nur sehr eingeschränkt tauglich sind. Für unseren Zweck bietet sich an, einfach Fakten als »positiv« oder »negativ« zu typisieren.

Ergänzt man dies noch mit der Angabe der Quelle, aus der das Faktum stammt, gelangen wir zu der folgenden Definition:

Definition 13.6 () Ein *Kontextfaktum* zu einem Kontextschema $C = (F, H)$ ist ein Tupel $(u, f, v, o, t_0, \text{valid}, \alpha_0, \text{source})$, wobei

- u ein Benutzer,

- $f \in F$ ein Kontextmerkmal
- $o \in \{=, \neq\}$ ein Indikator für positive bzw. negative Fakten,
- v ein Wert aus $\text{value-space}(f)$
- t_0 der Zeitpunkt ist, zu dem das Faktum der Informationsbasis hinzugefügt wurde (Transaktionszeitpunkt),
- valid ist ein Zeitintervall, innerhalb dessen Merkmalswert gültig ist und
- $\alpha_0 \in [0, 1]$ die Wahrscheinlichkeit zum Zeitpunkt t_0 darstellt, dass für den Benutzer U das Merkmal f den Wert v hat.
- source die einliefernde Quelle

13.2. Alterung

Bislang kann der Faktendienst nur Fakten *speichern*, die mit Unsicherheit und eingeschränkten Gültigkeitszeiträumen annotiert sind. Hierüber lassen sich noch keine Qualitätszusicherungen oder ein aktiver Umgang mit Imperfektion (wie KNF1 es fordert) realisieren, die Phänomene abfedern können wie z.B., dass wir nicht darüber informiert werden, dass bestimmte Werte nicht mehr gelten, gleichzeitig aber auch nicht unnötig Fakten als »veraltet« deklarieren wollen. Hier wird eine Nachbildung des »Alterungsprozesses« von Informationen benötigt, der die »natürliche« Einschätzung nachbildet, dass allmählich immer weniger wahrscheinlich ist, dass bestimmte Informationen noch mit der Realität übereinstimmen.

13.2.1. Existierende Ansätze zur Alterung

Dieses Problem wurde bislang beim Umgang mit Kontextinformationen kaum berücksichtigt. Dementsprechend existieren auch wenige Ansätze:

- Einfache Ansätze wie z.B. [MCF⁺94] benutzen Zeitfenster. Für die weitere Verarbeitung werden nur Fakten innerhalb eines bestimmten Zeitfensters benutzt. Teilweise wird dieses Zeitfenster auch innerhalb des jeweiligen Algorithmus adaptiv gewählt (so z.B. [WK96b]).
- KOYCHEV erarbeitete im Rahmen des sog. »Concept Drift«-Problems ein Verfahren, das ältere Fakten geringer gewichtet als neue Fakten [Koy00], um so ein »graduelles Vergessen« zu implementieren. Hierzu benutzt er lineare »Vergessensfunktionen« (Gewichtsfaktor in Abhängigkeit von der Zeit). Dabei wurde gezeigt, dass für

interessenbasierte Empfehlungen die Fehlerrate der Empfehlungen um 20 Prozent verringert werden konnte.

- Ähnlich schlagen WEBB UND KUZMYCZ [WK96a] im Bereich der Lernermodellierung das Konzept des sog. »Data Aging« vor. Szenario hierbei ist die möglichst präzise Abbildung des Kenntnisstandes des Lernenden, um so voraussagen zu können, ob der Lernende wahrscheinlich einen Fehler macht. Die Grundidee dabei ist, dass die Konfidenz älterer Indizien, die für das Mining von Assoziationsregeln benutzt werden, geringer gewichtet werden. Da hierbei von diskreten Zeitpunkten (in Form von Tests) ausgegangen wird, wird mit jeder »Runde« die Konfidenz mit einem festen Faktor reduziert. Die Evaluierungsergebnisse [WK98] zeigen, dass der pauschale Einsatz von Konfidenzabschlägen keine signifikanten Verbesserungen brachte; allerdings legen die Evaluierungsergebnisse nahe, dass ein differenzierterer Einsatz der Methode (unterschiedliche Behandlung von Indizien unterschiedlichen Typs) bessere Ergebnisse liefert.
- KREUZ [Kre00] baut auf den Erkenntnissen der Kognitionspsychologie auf und geht davon aus, dass die Relevanz von Fakten in einer Wissensbasis umgekehrt proportional zum Alter derselben ist und definiert auf der Basis eine Alterungsfunktion. Die Ergebnisse im Bereich der PC-Technologien sind hier vielversprechend.

Aus diesen Ergebnissen lässt sich schließen, dass ein Alterungsmechanismus in bestimmten Anwendungsfällen das Dilemma Vollständigkeit vs. Qualität/Korrektheit teilweise auflösen kann. Allerdings sind die bislang vorgestellten Verfahren in zwei Punkten unzureichend: (1) Sie können nicht die unterschiedliche Dynamik der Realwelt abbilden (alles wird gleich behandelt) und (2) das benutzte Modellinstrument zur Anwendung ist eng an die konkreten Algorithmen gekoppelt. Daraus ergibt sich, dass eine differenzierte Anwendung auf unterschiedliche Kontextfakten brauchen und wir die Alterung als weitgehend entkoppelt umsetzen müssen.

13.2.2. Repräsentation und Semantik von Alterung

Um diesen Forderungen Rechnung zu tragen, präzisieren wir im folgenden zunächst den Begriff der Alterung von Kontextinformationen. Informell ist hiermit gemeint, dass die Wahrscheinlichkeit, dass ein Faktum zutrifft, in Abhängigkeit vom Abstand zu dem Zeitpunkt, zu dem es gesammelt wurde, nach bestimmten Gesetzmäßigkeiten abnimmt. Damit wird anders als beim Zeitfensteransatz ein Faktum nicht zu einem bestimmten Zeitpunkt abrupt entfernt, sondern man erhält ein graduelles Vergessen. Stellen wir solche zeitabhängigen Wahrscheinlichkeit als generischen Dienst zur Verfügung, kann überdies jedes Verfahren für sich entscheiden, wie es diese Information benutzt, z.B. als Gewichte oder

durch einen Alpha-Schnitt nur Fakten berücksichtigt, die über eine Mindestwahrscheinlichkeit verfügen.

Formeller lässt sich das wie folgt definieren:

Definition 13.7 (Alterungsfunktion) Eine *Alterungsfunktion* ist eine Funktion $a : \text{TIME} \rightarrow [0, 1]$ mit den Eigenschaften:

- a ist stückweise stetig
- a ist monoton fallend

Sie gibt an, um welchen Faktor die initiale Wahrscheinlichkeit eines Faktums in Abhängigkeit von der Distanz zum Transaktionszeitpunkt des Faktums abnimmt.

Dabei ist klar, dass Alterungsfunktionen immer nur eine Annäherung an die Realität bleiben (wie auch das abrupte Vergessen und die unbegrenzte Gültigkeit als einfache Alternativstrategien).

Alterungsfunktionen sind auch nicht global gültig, sondern spezifisch für bestimmte Kontextmerkmale. Manche Informationen sind sehr stabil (z.B. persönliche Daten wie Name, Geburtsdatum), andere zumindest längerfristig (Abteilungszugehörigkeit), während Informationen wie die aktuelle Aufgabe, der aktuelle Stresszustand etc. sehr volatil sind. Deshalb annotieren wir Kontextmerkmale mit spezifischen Alterungsfunktionen

Definition 13.8 (Alterungsannotation) Ein *Alterungsannotation* auf einer Menge von Kontextmerkmalen F ist eine Funktion $A : F \rightarrow \{a \mid a \text{ ist eine Alterungsfunktion}\}$, die jedem Kontextmerkmal eine Alterungsfunktion zuordnet.

Damit lässt sich für ein Faktum $(u, f, v, o, t_0, \text{valid}, \alpha_0)$ also die Wahrscheinlichkeit zum Zeitpunkt t^* berechnen als

$$\alpha_{t^*} = [A(f)](t^* - t_0) \cdot \alpha_0$$

Was die Gestalt der Alterungsfunktionen anbelangt, so ist an dieser Stelle keine Einschränkung zu machen. Allerdings ist im Rahmen der Diskussion der Anfragebearbeitung zu untersuchen, welche Alterungsfunktionen aus Effizienzgründen zu bevorzugen sind.

13.3. Anfragen an den Kontextfaktendienst

In diesem Abschnitt geht es um die Anfragemöglichkeit, die für die Realisierung der Methode `getFacts` benötigt werden.

13.3.1. Grundstruktur und Semantik der Anfragen

Anfragen an den Kontextdienst müssen ein effizientes Filtern nach den Bestandteilen der Kontextfakten unterstützen. Hierzu gehören die Kontextmerkmale, die Benutzer, die Transaktionszeitpunkte, Gültigkeitszeiträume sowie die Konfidenz. In manchen Situationen kann auch eine Filterung nach den eigentlichen Werten sinnvoll sein, insbesondere bei benutzerübergreifenden Anfragen, so dass wir bei der folgenden Anfragegrundstruktur landen, die sich syntaktisch an SQL orientiert, allerdings spezielle einfache Konstrukte einführt:

```

SELECT|SUBSCRIBE POSITIVE|NEGATIVE|ALL FACTS FOR <featureURI>
[FOR USER          <userlist>]?
[VALID             <dateRange>]?
[ADDED            <dateRange>]?
[WITH CONFIDENCE  >= <confidence>]?
[WHERE            <valueCondition>]?
[ORDER BY        <field>]?

<userList>        -> <userURI> [, <userURI>]*
<dateRange>       -> ( [<date>|-INF], [<date>, INF] )
<confidence>     -> <double>
<valueCondition> -> <valueCondition> AND <valueCondition>
                  | valueCondition OR valueCondition
                  | NOT valueCondition
                  | valueConditionAtom
valueConditionAtom -> predicate valueExpression*

```

Hierbei bezeichnet:

- *SELECT|SUBSCRIBE* ob eine einmalige oder kontinuierliche Anfrage gestartet werden soll
- *VALID* den Gültigkeitszeitraum
- *ADDED* den Transaktionszeitraum
- *WITH CONFIDENCE* die minimale Konfidenz zu Beginn des Gültigkeitszeitraumes
- *WHERE* eine Bedingung für die Werte der Kontextfakten
- *ORDER BY* die gewünschte Ordnung

Ergebnis einer solchen Anfrage ist eine Menge von Fakten; bei der Auswertung werden die einzelnen Abschnitte konjunktiv verknüpft. Dabei können zwei unterschiedliche Interaktionsparadigmen unterstützt werden: Bei »SELECT« werden klassische synchrone Anfrage-Antwort-Interaktionen unterstützt, während bei »SUBSCRIBE« die asynchrone Benachrichtigung möglich wird.

Werden einzelne Abschnitt nicht angegeben, so werden Standardwerte eingesetzt:

- *FOR USER*: die Menge aller Benutzer
- *VALID*: den aktuellen Zeitpunkt
- *ADDED*: $(-\text{INF}, \text{INF})$
- *WITH CONFIDENCE*: 0
- *WHERE*: true

Formell lässt sich die Anfragesemantik dann wie folgt definieren:

Definition 13.9 (Anfrage an Kontextfaktendienst und deren Semantik) *Eine Anfrage q ist ein Tupel*

$$q = (U^*, F^*, \text{valid}^*, T^*, \text{condition}, O^*, \text{alpha}^*), \text{ wobei}$$

- U^* eine Menge von Benutzern
- F^* eine Menge von Kontextmerkmalen
- valid^* ein Gültigkeitsintervall (wobei t^* im folgenden als Bezeichner für den Intervallbeginn verwendet wird),
- T^* ein Intervall für den Transaktionszeitpunkt,
- condition eine Wertbedingung darstellt,
- $O^* \subseteq \{=, \neq\}$ angibt, ob positive und/oder negative Fakten geliefert werden sollen
- alpha^* eine Mindestkonfidenz angibt

Damit lässt sich die Semantik der Anfrage angewandt auf eine Menge von Kontextfakten M beschreiben als

$$\begin{aligned} \text{result}(q, M) := \{ & m = (u, f, v, o, t_0, \text{valid}, \alpha_0, \text{source}) \mid \\ & u \in U^*, t_0 \in T^*, o \in O^*, \\ & \exists f' \in F^* : f' \triangleright f, \text{valid} \cap \text{valid}^* \neq \emptyset, \\ & \text{eval}(v, \text{condition}) = \text{true}, \alpha_{t^*} \geq \alpha^* \} \end{aligned}$$

wobei $\alpha_{t^*} = [A(f)](t^* - t_0) \cdot \alpha_0$ die oben eingeführte, mit Hilfe der Alterungsfunktionen berechnete Konfidenz zum Zeitpunkt t^* darstellt.

Was jetzt noch offen ist, ist die Auswertungsfunktion $eval(v, condition)$. Da sie mit einer zweiwertigen Logik arbeiten kann, ist die Auswertung komplexer Ausdrücke unmittelbar einsichtig. Auch auf atomarer Ebene kann einfach auf die entsprechenden Vergleichsoperatoren der zugrundeliegenden Datentypen zurückgegriffen werden. Einzig der Umgang mit Wertebereichen aus der Ontologie (Anforderung KF6) bedarf gesonderter Betrachtung, was im nächsten Abschnitt erfolgen soll.

13.3.2. Anfragen mit Ontologien auf Datentypebene

Auch wenn wir uns oben aus Gründen der Anfragebearbeitung gegen einen Ansatz entschieden haben, der nur auf OWL-Reasoning setzt, ist die Bereitstellung adäquater Anfragemechanismen für Elemente aus der Domänenontologie essentiell, wie in Anforderung KF6 festgehalten ist.

Die verfolgte Grundidee ist, dass wir Ontologien als Datentyp auffassen, so dass die Menge der Instanzen der Ontologie dem Wertebereich bestimmter Kontextmerkmale (die im folgenden oft als »ontologiewertige« Merkmale bezeichnet werden) entspricht.¹ Neben dem Wertebereich benötigen wir hierfür noch Prädikate, die im Rahmen der Anfragebedingungen benutzt werden können und das in der Ontologie explizierte Hintergrundwissen nutzbar machen.

Naheliegender Ansatz ist hier, sich bei existierenden Ontologieanfragesprachen zu bedienen. Neben OWL-QL [FHH05], der SIRUP Anfragesprache [ZSD05] und anderen Ansätzen bietet sich hier vor allem SPARQL an, das bereits in Teil III benutzt wurde. SPARQL-Anfragen können als Unteranfragen in Faktenanfragen eingebunden werden:

```
VALUE MATCHES (?x) IN { SPARQL-Anfrage }
```

Allerdings macht dies »typische« einfache Anfragen umständlich zu formulieren, so dass alternativ spezielle Prädikate eingeführt werden sollten, die zudem durch ihre eingeschränkte Mächtigkeit im Rahmen der Anfragebearbeitung Optimierungsspielräume eröffnen. »Typische« Anfragen sind dabei die Überprüfung, ob eine Instanz zur Extension eines bestimmten Konzeptes gehört und die Überprüfung, ob eine Instanz zu einer anderen Instanz in einer bestimmten semantischen Beziehung steht. Hierfür werden die beiden Prädikate *instanceOf(conceptURI)* und *relatedToConceptInstance(propertyURI,conceptURI)* sowie *relatedToInstance(propertyURI,instanceURI)* eingeführt. Ihre Semantik lässt sich als SPARQL-Anfrage ausdrücken (?x bezeichnet jeweils den zu prüfenden Wert):

¹Es wird hier nicht betrachtet, dass für bestimmte Kontextmerkmale nur Instanzen bestimmter Konzepte sinnvoll sind.

```
?x instanceOf(c) := ASK { ?x rdf:type c }
```

```
?x relatedToConceptInstance(?p, ?c) :=  
  ASK { {?x p ?y}, {?y rdf:type c} }
```

```
?x relatedToInstance(?c, ?i) :=  
  ASK {?x p i}
```

13.4. Anfragebearbeitung auf der Basis relationaler Datenbanksysteme

Besonders vor dem Hintergrund, dass wir gemäß Anforderung KF3 große Datenmengen zu erwarten haben, bei denen aufgrund der unterschiedlichen Dynamik auch nicht einfach festlegen können, welcher Zeitraum für Anfragen interessant ist, ist auf eine effiziente Anfragebearbeitung auch bei vielen Kontextfakten zu achten (KNF4). Hier bieten sich relationale Datenbanksysteme als etablierte Systeme mit effizienten Anfragebearbeitungsverfahren an, sofern wir diese für unsere Zwecke nutzen können. In diesem Abschnitt soll daher analysiert werden, ob und in welcher Form die Anfragen an den Kontextfaktdienst mit Hilfe der Anfragen an das relationale Datenbanksystem beantwortet werden können.

Eine Alternative könnten die entstehenden skalierbaren Triple-Stores für RDF-Daten sein. Allerdings ist das verwendete Kontextmodell nicht RDF-konform und erfordert zusätzlich zur Subjekt-Prädikat-Objekt-Struktur zwingend weitere Tupelmetadaten, die sich zwar in sog. Quadruple-Stores per Reifikation abbilden lassen, was allerdings umständlich ist und zulasten der Performanz geht. Weiterhin kommen Triple-Stores nur schlecht mit Unsicherheit und negativen Fakten bei der Anfragebearbeitung zurecht, da diese Modellelemente nicht in RDF enthalten sind. Deshalb macht an dieser Stelle ein eigener Ansatz Sinn, der sich direkt auf relationale Datenbanktechnologie abstützt (die oft auch den Triple-Stores zugrundeliegt).

Betrachtet man das Datenmodell, das wir für Kontextfakten definiert haben, so lässt sich dies durch seine bewusst beschränkte Ausdrucksmächtigkeit relativ leicht in das relationale Modell übersetzen, da Kontextfakten Tupel darstellen, die sich in einer Relation speichern lassen. Aus praktischen Gründen (wie z.B. unterschiedliche Datentypen für die Werte) werden die Tupel allerdings auf mehrere Relationen verteilt. Ein ausführliches relationales Schema (für Oracle 10i) findet sich in Anhang B.

Die in Abschnitt 13.3 eingeführte Anfragesprache für den Zugriff auf den Kontextfaktdienst lässt sich so größtenteils unmittelbar in Anfragebedingungen übersetzen, die sich effizient durch Indexstrukturen auswerten lassen. Hierzu gehört die Filterung nach Benutzern, Gültigkeitsbereich, Transaktionsbereich sowie positiv/negativ. Auf den ersten Blick

erscheint die Berücksichtigung der Vererbung bei Kontextmerkmalen nicht ganz so trivial; allerdings lässt sich durch Ersetzen eines Kontextmerkmals f durch alle $f_i \triangleleft f$ der Effekt leicht erreichen. Zudem lässt sich dies problemlos – da wir von einem relativ statischen Kontextfaktenschema ausgehen – vorausberechnen.

Die einzigen Schwierigkeiten für eine effiziente Anfrageauswertung sind (1) die Berechnung der Alterungsfunktionen und (2) die Auswertung der Wertbedingungen im Falle von ontologiewertigen Kontextmerkmalen. Diese beiden Aspekte sollen im folgenden vertieft werden.

13.4.1. Auswertung der Alterungsfunktionen

Anfragen mit einer Mindestkonfidenz α zum Referenzzeitpunkt t müssen die folgende Anfragebedingung für jedes Tupel auswerten:

$$\alpha_t = a(t - t_0) \cdot \alpha_0 \geq \alpha^*$$

Dies erfordert jeweils den kompletten Scan der Faktenbasis. Ziel muss hier sein, durch den Einsatz von Indexstrukturen den Berechnungsaufwand zu minimieren. Dies lässt sich dadurch erreichen, dass wir die obige Ungleichung so umformen, dass auf einer Seite der Ungleichung nur t_0 und α_0 auftauchen. Dann lässt sich dieser Ausdruck vorausberechnen (z.B. direkt beim Einbringen in die Faktenbasis). Nehmen wir an, dass sich $a(t - t_0)$ zerlegen lässt in $f(t) \cdot g(t_0)$, dann erhalten wir:

$$\begin{aligned} (f(t) \cdot g(t_0)) \cdot \alpha_0 &\geq \alpha^* \\ g(t_0) \cdot \alpha_0 &\geq \frac{\alpha^*}{f(t)} \end{aligned}$$

Die obige Umformung gilt für $0 < f(t) \leq 1$.

Damit lässt sich die linke Seite wie gewünscht für jedes vorausberechnen, da sie unabhängig vom Referenzzeitpunkt t ist. Die rechte Seite muss für eine Anfrage ebenfalls nur ein einzelnes Mal (pro Alterungsfunktion und damit pro Kontextmerkmal) berechnet werden. Damit lässt sich auf dem vorausberechneten Attribut mit dem Wert $g(t_0) \cdot \alpha_0$ ein Index für Bereichsanfragen anlegen und nutzen.

Welche Klassen von Alterungsfunktionen lassen sich wie gewünscht zerlegen? Eine naheliegende Funktion ist hier für $a(x) := e^{rx+s}$. Diese ist für $r < 0$ wie gefordert monoton fallend. Damit ist

$$a(t - t_0) = e^{r(t-t_0)+s} = e^{rt} e^{rt_0+s}$$

Dies ist eine Zerlegung mit $f(t) = e^{rt}$ und $g(t_0) = e^{rt_0+s}$, wobei auch $0 < f(t) \leq 1$ gilt.

Damit bieten sich Exponentialfunktionen der Struktur e^{rx+s} als Alterungsfunktionen an. Sie entsprechen auch einer Verstetigung des Vorgehens von WEBB UND KUZMYCZ sowie Modellen des Vergessens des menschlichen Gehirns [BN06].

13.4.2. Auswertung der Ontologieoperatoren

Was jetzt noch bleibt, ist die Frage, wie man Bedingungen auf ontologiewertigen Merkmalen effizient auswertet. Hier existieren grundsätzlich drei Lösungsmöglichkeiten (vgl. Abb. 13.1:

- **Inferenzmechanismen im Datenbanksystem integriert.** Hierbei wird direkt im relationalen Datenbanksystem ein Datentyp Ontologie angelegt, wie es z.B. von Oracle in [DCES04] und [CDES05] vorgeschlagen wird. Dies ist sicherlich die komfortabelste Lösung, ist allerdings derzeit spezifisch für ein RDBMS. Weiterhin wird hier bislang aus Effizienzgründen von einem stark eingeschränkten Reasoning ausgegangen.
- **Einbindung von relationalen Daten als extensionale Prädikate.** Bei Reasonern, die auf Datalog aufsetzen (wie z.B. KAON2, vgl. [Mot06]) besteht die einfache Möglichkeit, Relationen in RDBMS als extensionale Prädikate einzubinden. Dadurch werden auch Operationen auf Datentypen (z.B. Vergleiche auf numerischen Daten oder Datumsangaben) leicht realisierbar. Allerdings sind die derzeitigen Implementierungen dahingehend eingeschränkt, dass sie kein sinnvolles »query shipping« unterstützen, also das Durchreichen von möglichst vielen Anfragebedingungen an das zugrundeliegende RDBMS, um so möglichst wenig Tupel zurückzuhalten. Dies liegt im wesentlichen an den verwendeten Reasoning-Algorithmen, die dies nicht erlauben, so dass große Datenmengen in den Hauptspeicher geladen werden müssen.
- **Query Rewriting.** Für beschränkte Anwendungsfälle bietet es sich auch an, die Reasoning-Semantik aus der Ontologie durch Umschreiben der Anfrage umzusetzen. Dies setzen ERDMANN UND DECKER auch für XML bereits in [ED00] um.

Für die beschränkte Form von Reasoning, wie wir sie im Rahmen der Definition der Anfragesprache eingeführt haben, reicht der Rewriting-Ansatz aus, wenn man ihn so flexibel gestaltet, dass der Zeitpunkt des Zugriffs auf den Reasoner offen bleibt. Dies bedeutet, dass man die Flexibilität hat, bei Eintreffen der Anfrage direkt die Ontologiebedingung in eine Bedingung für das relationale Datenbanksystem zu übersetzen oder zunächst die Anfrage ohne die Wertbedingung auszuführen und anschließend die Ergebnismenge mit Hilfe des Reasoners zu filtern. Ein Beispiel ist die Ausführung des *instanceOf*-Prädikates. Die erste Variante expandiert das Konzept durch seine Extension; die zweite Variante überprüft

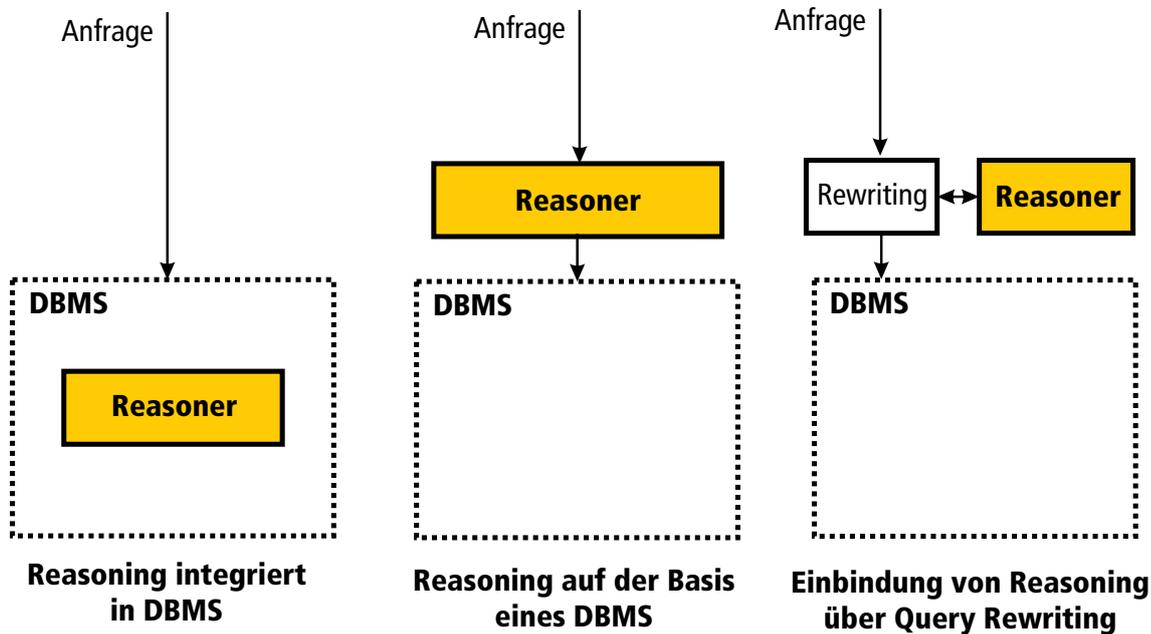


Abbildung 13.1.: Alternativen zur Kombination von Reasoning und RDBMS

die Elemente der Ergebnismenge, ob der Wert die Ontologiebedingung erfüllt. Dadurch ergeben sich Optimierungsspielräume wie bei der Join-Reihenfolge, die man unter Zuhilfenahme der Abschätzung der Selektivität nutzen kann. Hierzu kann man beispielsweise die Anzahl der Instanzen eines Konzeptes heranziehen.

13.5. Kontinuierliche Anfragen

Anforderung KF2 fordert neben der synchronen Interaktion mit dem Kontextverwaltungsdienst auch die Möglichkeit, asynchron benachrichtigt zu werden, um so auch proaktive Funktionalität einfach anbieten zu können. Zusätzlich sieht die Anbindung von Anreicherungsagenten über die Blackboard-Architektur ebenfalls im Kern Benachrichtigungen über Änderungen vor. Sicherlich ließe sich diese Funktionalität auch durch periodisches Polling realisieren; allerdings steigt hier die Last für den Kontextfaktdienst erheblich an, wenn jeweils komplette Anfragen ausgeführt werden müssen (vgl. [RPS06]). Weiterhin ist der Verarbeitungsaufwand auf Seiten der aufrufenden Komponenten ebenfalls höher, da diese entweder jede Ergebnismenge daraufhin analysieren müssen, ob neue Fakten vorliegen bzw. alte nicht mehr die Mindestkonfidenz aufweisen (wenn sie ihr Verhalten auf Änderungen abstimmen), oder die Zeitfenster für ihre Anfragen anpassen und sich selbst

um die Effekte der Alterung kümmern.

Was benötigt wird, sind »kontinuierliche Anfragen« (*continuous queries*) [BW01]. Da es sich bei dem Kontextfaktendienst um eine sog. »append-only«-Datenbank handelt, bei der nur neue Tupel hinzukommen, aber keine bestehenden gelöscht oder verändert werden, sind hier eigentlich kontinuierliche Anfragen sehr einfach zu implementieren, da diese stets monoton sind [TGNO92]. Allerdings werden durch die Alterung die Anfragen nicht-monoton: Ergebnisse zum Zeitpunkt t_1 erfüllen Zeitpunkt $t_2 > t_1$ die Anfragebedingung bzgl. der Konfidenz nicht mehr. Damit können nicht einfach neue Tupel ergänzt werden; es müssen alte Tupel aus der Ergebnismenge entfernt werden.

Grundsätzlich müssen also Ergebnisströme von kontinuierlichen Anfragen sowohl neue Fakten liefern als auch bereits gelieferte Fakten löschen. Ersteres lässt sich einfach bei Eintreffen neuer Fakten durch Überprüfen der Anfragebedingung testen. Da das Halten von Zustandsinformationen über gelieferte Ergebnisse aufwendig ist, ist es sinnvoller, für zweiteres für jedes Ergebnis ein Auslaufdatum zu berechnen, das maximal am Ende des explizit angegebenen Gültigkeitszeitraum liegt. Liegt ein solches Auslaufdatum vor, so kann der jeweilige Klient selbst die Ergebnisse einfach zum berechneten Zeitpunkt bereinigen.²

Wie lässt sich nun dieses Auslaufdatum berechnen? Hier kann man wieder auf die Konfidenzbedingung zurückgreifen. Gesucht ist nun der Zeitpunkt t , zu dem die Mindestkonfidenz α^* erreicht ist.

$$\begin{aligned} \alpha_t &= a(t - t_0) \cdot \alpha_0 = \alpha^* \\ a(t - t_0) &= \alpha^* / \alpha_0 \end{aligned}$$

Dies lässt sich nun mittels numerischer Verfahren (Nullstellenverfahren) oder einfachen Wertetabellen approximativ lösen. Für die oben im Rahmen der relationalen Umsetzung favorisierten Alterungsfunktionen der Struktur e^{rx+s} lässt sich das auch exakt berechnen:

$$\begin{aligned} e^{r(t-t_0)+s} &= \alpha^* / \alpha_0 \\ r(t - t_0) + s &= \ln \alpha^* / \alpha_0 \\ t &= \frac{\ln(\alpha^* / \alpha_0) - s}{r} + t_0 \end{aligned}$$

Dieses Berechnungsergebnis lässt sich den gelieferten Ergebniselementen als Auslaufdatum hinzufügen.

²Hierfür lassen sich selbstverständlich generische clientseitige Komponenten bereitstellen, die diese Aufgabe übernehmen.

13.6. Fazit

Mit dem Kontextfaktendienst ist nun die Grundlage geschaffen für eine angemessene Kontextinfrastruktur, die mit Unsicherheit und Dynamik umgehen kann und dennoch skalierbar ist. Hierbei spielt neben der durchgängig temporalen Perspektive vor allem die Alterung eine große Rolle, die Vollständigkeit und Qualität der über einen Benutzer verfügbaren Informationen erhöhen kann. Im nächsten Kapitel geht es nun darum, die Widersprüche und Konflikte aufzulösen, die innerhalb der Fakten existieren können.

14.

Kontextinformationsebene

Im Gegensatz zur Kontextfaktenebenen werden auf der Kontextinformationsebene nach außen hin im wesentlichen nur Kontextmerkmale und ihre Werte zu einem bestimmten Zeitpunkt (in der Regel der aktuelle) sichtbar. Die einzelnen Werte werden dabei hinsichtlich ihrer Gewissheit mit Konfidenzen versehen. Es existieren hier keine Konflikte im Sinne von Widersprüchen, wobei allerdings die Übereinstimmung mit der realen Welt (also die Situation) nicht gewährleistet ist.

Der Kontextinformationsdienst unterstützt die folgende Schnittstelle:

Kontextinformationsdienst
<pre>UserContext getContextForUser(ContextQuery) UserContext getUsers(UserQuery) TruthValue check(ContextCheckQuery) void registerQuery(ContextQuery, Callback) void unregister(Callback) void addContextFact(ContextFact)</pre>

14.1. Schema für Kontextinformationen

Wichtiger Unterschied zum Umgang mit Kontextfakten im vorangegangenen Kapitel ist es, dass wir nun mehr Zusicherungen bzgl. der Qualität der gelieferten Informationen geben wollen. Für viele Anwendungen ist der Umgang mit widersprüchlichen Daten aufwendig, so dass wir dies auf dieser Ebene aufzulösen versuchen. Hierzu ist es allerdings notwendig, klar zu definieren, was Anwendungen als Ergebnisse erwarten können. Dies

geschieht üblicherweise mittels einer Schemadefinition, die wir gegenüber der Faktenebene erweitern müssen, um Widersprüche auszuschließen:

Definition 14.1 (Kontextinformationsschema) Ein *Kontextinformationsschema* ist ein Tupel (FS, m) , wobei $FS = (F, H)$ ein Kontextfaktenschema darstellt und $m : F \rightarrow \{1, 2, \dots\}$ eine Funktion darstellt, die für ein Merkmal festlegt, ob zu einem bestimmten Zeitpunkt einer oder mehrere Werte zulässig sind. Damit m verträglich ist zur Merkmalshierarchie, muss gelten:

$$\forall f_1 \triangleright f_2 : m(f_1) \geq m(f_2)$$

Darauf aufbauend lassen sich nun die Instanzen definieren, die sich von den Kontextfakten dadurch unterscheiden, dass sie nicht mehr die temporalen Aspekte exponieren. Hauptgrund hierfür ist, dass wir davon ausgehen, dass hier mehrere Fakten als Indizes aggregiert werden, so dass insbesondere die Transaktionszeitpunkte keinen Sinn mehr ergeben.

Definition 14.2 (Kontextmerkmalswert) Ein *Kontextmerkmalswert* ist ein Tupel (u, f, o, v, α) , wobei

- u ein Benutzer
- f ein Kontextmerkmal,
- $o \in \{=, \neq\}$ ein Indikator für positive bzw. negative Fakten,
- $v \in \text{value-space}(f)$ den Wert darstellt.
- Die Konfidenzwahrscheinlichkeit α gibt an, dass mit der Wahrscheinlichkeit α aktuell der Benutzer u für das Merkmal f den Wert v hat (für $o ='$ bzw. nicht den Wert v hat (für $o ='\neq'$)).

14.2. Aggregation von Fakten und Konfliktauflösung

Wie können wir nun auf der Basis der Kontextfaktenschemen, wie wir sie durch Anfragen an den Kontextfaktendienst erhalten, Kontextinformationsmengen erzeugen? Dies ist die Aufgabe der Konfliktauflösung (die auch die einfache Aggregation mehrfacher Werte umschließt). Sie lässt sich grundlegend in zwei Problemtypen aufteilen. Auf der einen Seite stehen Kontextmerkmale, die nur eine Ausprägung zulassen. Hier ist das Erkennen eines Konfliktes relativ einfach, und bei der Auflösung des Konfliktes geht es »nur« um die Abwägung von unterschiedlichen Alternativen gegeneinander. Unübersichtlicher ist die Lage bei mehrwertigen Merkmalen, bei denen die Konflikterkennung nicht so einfach ist, weil – bis auf den Spezialfall des gleichzeitigen Vorhandenseins von positiven und negativen

Fakten – keine unmittelbaren Indikatoren zur Verfügung stehen. Im folgenden wird daher zunächst das Problem für einwertige Merkmale, anschließend für mehrwertige Merkmale thematisiert.

14.2.1. Begriff des Konflikts

Auf den ersten Blick scheint klar, dass in den von unterschiedlichen Verfahren ermittelten Kontextfakten Konflikte auftauchen können. Allerdings ist nicht so unmittelbar greifbar, was denn genau als Konflikt aufzufassen ist:

- **Positive und negative Fakten.** Klar ist, dass ein Konflikt vorliegt, wenn in einer Menge von Fakten für ein Kontextmerkmal ein bestimmter Wert einmal in einem positiven Faktum und einmal in einem negativen Faktum enthalten ist.
- **Verletzung von Kardinalitäten.** Ebenso liegt ein Konflikt vor, wenn in einer Faktmenge mehr unterschiedliche Werte enthalten sind als entsprechend der Kardinalitäten zulässig sind. Kann ein Merkmal nur eine Ausprägung zu einem bestimmten Zeitpunkt haben, so können nicht mehrere Werte zutreffen.
- **Redundante Fakten.** Nicht ganz so intuitiv einsichtig ist der Konflikt, der durch das mehrfache Vorkommen desselben Wertes für ein Kontextmerkmal entsteht. Auch diese Situation ist problematisch, da Anwendungen üblicherweise nur ein Vorkommen erwarten und jeweils (potentiell) unterschiedliche Konfidenzinformation vorliegen.

Damit lässt sich der Konfliktbegriff wie folgt formalisieren:

Definition 14.3 (konfliktfrei) Eine Menge M von Kontextmerkmalswerten ist **konfliktfrei** bzgl. eines Kontextinformationsschemas, wenn $\forall k = (u, f, o, v, \alpha) \in M$ gilt, dass kein $k' = (u', f', o', v', \alpha') \in M$ existiert mit $k \neq k'$, $u = u'$ sowie $f' \triangleleft f$ mit einer der folgenden Konfliktsituationen:

1. $o = o'$ und $v = v'$ (keine doppelte Werte)
2. $o \neq o'$ und $v = v'$ (keine widersprüchlichen Werte)

Weiterhin muss gelten:

$$|\{k' \neq k \in M \mid o' = o, f' \triangleleft f\}| \leq m(f)$$

d.h. es dürfen nicht mehr Werte für ein Merkmal vorhanden sein als das Schema zulässt.

14.2.2. Konfliktauflösungsstrategien

Wesentliche Funktion der Kontextinformationsebene ist die Auflösung von Konflikten, so dass nutzende Anwendungen konfliktfreie Ergebnismengen erhalten. Dies ist die Aufgabe von Konfliktoperatoren:

Definition 14.4 (Konfliktauflösung) *Ein Konfliktauflösungsoperator ist eine Funktion, die eine Menge C von Kontextfakten in eine konfliktfreie Menge von Kontextmerkmalswerten M überführt.*

Da ein Konfliktoperator bei Konflikten keine Überprüfung in der Realwelt vornehmen kann, muss dieser mit heuristischen Verfahren arbeiten, die die in den Kontextfakten enthaltenen Metadaten benutzen. Hier können unterschiedliche Strategien zum Einsatz kommen, die im folgenden untersucht werden. HECKMANN [Hec06b] hat beispielsweise vier grundsätzliche Strategien herausgearbeitet:

- CR MostRecent.** Hierbei wird bei im Konflikt stehenden Fakten dem jüngeren Priorität eingeräumt. Die dahinterstehende Annahme ist hierbei, dass die unterschiedlichen Kontextgewinnungsverfahren im wesentlichen zuverlässige Ergebnisse liefern, so dass dem jüngsten Ergebnis am meisten zu vertrauen ist.
- CC MostConfident.** Hierbei wird das Faktum mit der derzeit höchsten Konfidenz gewählt. Das bedeutet, dass man hier von Verfahren mit ungewissen Ergebnissen ausgeht, wobei allerdings ihre Einschätzung der Gewissheit zuverlässig ist, so dass das zuverlässigste Ergebnis auch am wahrscheinlichsten ist.
- CN MostNamed.** Hierbei werden gleichlautende Fakten aggregiert und dann das mit der höchsten Anzahl ausgewählt. Diese Strategie macht nur Sinn, wenn (a) man von mehreren (statistisch) unabhängigen Verfahren oder (b) von (statistisch) unabhängigen Ergebnissen eines einzelnen Verfahrens ausgeht.
- CS MostSpecific.** Sind die in Konflikt stehenden Fakten auf unterschiedlichen Abstraktionsebenen, so wird dem spezielleren der Vorzug gegeben. Hier steckt die Annahme dahinter, dass wenn ein Verfahren eine spezifischere Aussage machen kann, dass dann sein Ergebnis auch zuverlässiger ist. Hier lassen sich zwei Untervarianten identifizieren:
 - CSF** Es wird das speziellere Merkmal ausgewählt.
 - CSV** Es wird (bei ontologiewertigen Merkmalen) die speziellere Instanz ausgewählt.

Bei diesen Strategien werden die Elemente Transaktionszeitpunkt, Konfidenz sowie Kontextmerkmal bzw. Wert ausgenutzt. Zusätzlich lässt sich mit dem Gültigkeitsdatum noch eine weitere Strategie identifizieren:

CV LongestValid. Hierbei wird denjenigen Fakten der Vorzug gegeben, die den am weitesten in die Zukunft reichenden Gültigkeitszeitraum aufweisen. Dies macht nur Sinn bei Kontextquellen, die vorwiegend Ergebnisse mit bekannt eingeschränktem Gültigkeitsbereich liefern. Hier lassen sich auch zwei Untervarianten definieren:

CVV Als Ende des Gültigkeitszeitraumes wird das Intervallende von *valid* genommen.

CVD Als Ende des Gültigkeitszeitraumes wird der kleiner Werte aus Intervallende von *valid* und dem Ausfalldatum genommen.

Wie sieht nun der generelle Ablauf einer Konfliktlösungsstrategie aus? Sie lässt sich in drei Schritte aufteilen: (1) (optional) Zusammenfassen von Fakten (z.B. von wiederholten Fakten) und ggf. Hinzufügen von zusätzlichen Metadaten (wie z.B. Anzahl), (2) Sortieren der Fakten nach dem Kriterium, dem Priorität eingeräumt wird und (3) schrittweiser Aufbau der Ergebnismenge mit Hinzufügen von Fakten, solange die Menge widerspruchsfrei bleibt. Dies lässt in folgendem Algorithmenschema festhalten:

```
Set<ContextFeatureValue> resolveConflicts(List<ContextFact> facts)
{
    facts = strategy.aggregate(facts);
    facts = strategy.sort(facts);

    HashMap<Object,ContextFeatureValue> m;
    int countPositive = 0;

    foreach (f in facts)
    {
        // is the value already in the result set (positive or negative)
        if (! m.containsKey(f.value) )
        {
            if (f.factType = positive)
            {
                // is maximum cardinality already reached?
                if (countPositive < f.feature.maxCardinality)
                {
                    m.put(f.value,createFeatureValue(f));
                    countPositive++;
                }
            }
            else
                m.put(f.value,createFeatureValue(f));
        }
    }
}
```

```

return m.values;
}

```

Die oben eingeführten unterschiedlichen Strategien können anhand des Algorithmenschemas wie in folgender Tabelle dargestellt beschreiben:

	Aggregation	Sortierung
CR	-	$t_0 \downarrow$
CC	-	$\alpha^* \downarrow$
CN	Äquivalenzklassen (o, v) und deren Häufigkeit h Konfidenz als Maximum	$h \downarrow$
CSF	-	$f \downarrow$ (Basis H)
CSV	-	topologische Sortierung nach Klassen und deren Generalisierung
CVV	-	$valid_{max} \downarrow$
CVD	-	$\min(valid_{max}, t_{dropout}) \downarrow$

Tabelle 14.1.: Einordnung der Konfliktauflösungsstrategien in das Algorithmenschema

Die Strategien lassen sich dabei auch kombinieren, indem man aus unterschiedlichen Einzelkriterien (ggf. auch auf voraggregierten Fakten) eine gewichtete Summe berechnet, die dann als Sortierkriterium dienen.

14.3. Anfragen

Während wir beim Kontextfaktendienst von einer Art von Anfragen ausgegangen sind (mit den zwei Varianten synchron vs. asynchron), fordert KF1 drei unterschiedliche Anfragemöglichkeiten: nach einer Kontextinformationsmenge (a), nach der Überprüfung einer bestimmten Aussage (b) und nach Benutzern mit gemeinsamen Kontextinformationen (c). Diese unterschiedlichen Anfragemöglichkeiten müssen nun vom Kontextinformationsdienst auf der Basis der Möglichkeiten des Kontextfaktendienstes bereitgestellt werden – jeweils in einer synchronen und asynchronen Variante.

Diese unterschiedlichen Anfragearten (ContextQuery, CheckQuery und UserQuery) aus der Schnittstellendefinition und den Anforderungen sollen in den folgenden Abschnitten näher untersucht werden.

14.3.1. Anfragen nach Kontextmerkmalsausprägungen (*ContextQuery*)

Bei diesem Anfragetyp wird als Ergebnis eine Menge von Kontextmerkmalswerten erwartet. Die Anfrage hat also abstrakt die Struktur

$$getValues : (U^*, F^*, valid^*, T^*, \alpha^*, condition) \mapsto M$$

was anschaulich bedeutet: *Gib zu einer Menge von Merkmalen F^* für eine Menge von Benutzern U^* alle Merkmalswerte, die innerhalb des Zeitraums $valid^*$ mit einer Konfidenz α^* gelten und innerhalb des Zeitintervalls T^* hinzugefügt wurden.* Das lässt sich analog zur Einführung einer Anfragesprache im vorangegangenen Kapitel wie folgt in einer SQL-artigen Syntax darstellen:

```
SELECT | SUBSCRIBE POSITIVE | NEGATIVE | ALL INFORMATION
FOR          <featureURI>
FOR USER    <userlist>
[VALID      <dateRange>]
[ADDED      <dateRange>]
[WITH CONFIDENCE >= <confidence>]
[WHERE      <valueCondition>]
[RESOLVE CONFLICTS USING <strategy>]
```

Wie man bereits an der Ähnlichkeit Anfragestruktur zu den Anfragen an den Kontextfaktendienst erahnen kann, ist die Anfragebearbeitung in diesem Fall einfach. Die Anfragen können mit einer Einschränkung einfach durchgereicht werden: Die Anfrage an den Kontextfaktendienst muss – unabhängig ob nur positive oder negative Fakten gewünscht sind – alle Fakten anfordern, da ansonsten die Widersprüche in der Faktenbasis nicht aufgelöst werden können.

Die so erhaltenen Ergebnisse werden mit den im vorangegangenen Abschnitten vorgestellten und in der Anfrage angebbaren Konfliktauflösungsstrategie aggregiert und bereinigt und ggf. nach positiven oder negativen Aussagen gefiltert.

14.3.2. Überprüfung der Gültigkeit von Fakten (*CheckQuery*)

Dieser Anfragetyp zielt darauf ab, nicht nur vorhandene Fakten abzufragen, sondern die Faktenlage danach zu analysieren, ob ein bestimmter Merkmalswert »möglich« ist. Hier reicht keine binäre Sicht, sondern es werden zusätzliche Werte benötigt. Neben der Aussage *wahr(t)* (die Faktenlage stützt die Aussage explizit) und *falsch(f)* (die Faktenlage stützt das Gegenteil) werden auch die Wahrheitswerte *möglich/unbekannt(u)* (die Faktenlage stützt weder die Aussage noch ihr Gegenteil) und *widersprüchlich(i)* benötigt (die Faktenlage stützt sowohl die Aussage als auch ihr Gegenteil). Dies ist das Ergebnis dessen,

dass wir (a) bewusst in der Kontextfaktenbasis Widersprüche zugelassen haben und (b) die Kontextermittlung immer unvollständig bleibt (und wir damit von einer offenen Welt ausgehen müssen).

Die restliche Anfragestruktur ist sehr ähnlich:

$$checkValue : (U^*, F^*, valid^*, T^*, \alpha^*, condition) \mapsto \{t, f, u, i\}$$

Dies schlägt sich auch in der entsprechenden SQL-ähnlichen Syntax nieder:

```
CHECK INFORMATION
FOR                <featureURI>
FOR USER           <userlist>
[VALID             <dateRange>]
[ADDED            <dateRange>]
[WITH CONFIDENCE  >= <confidence>]
WHERE              <valueCondition>
```

Für den Umgang mit den vier Wahrheitswerten bei der Auswertung der *valueCondition* wird eine vierwertige Logik benötigt (vgl. zum Überblick [BD01]). ins Spiel.

Die Auswertung der Anfrageatome (*pred, arg*) erfolgt nach folgendem Schema: Sei hierzu C die Ergebnismenge der Anfrage $q = (U^*, F^*, valid^*, T^*, true, =, \neq, alpha^*)$ an den Faktendienst; dabei wird die Wertbedingung nicht durchgereicht, da ansonsten Widersprüche nicht erkannt werden können. Seien die Mengen M_{true}^+ der zutreffenden positiven, M_{false}^+ der nicht zutreffenden, M_{true}^- der widersprüchlichen negativen Fakten sowie M_{false}^- der nicht widersprüchlichen Fakten wie folgt definiert:

$$\begin{aligned} M_{true}^+ &:= \{c = (U, f, v, o, t, valid, \alpha) \in C \mid o = '=' \wedge pred(v, arg) = true\} \\ M_{false}^+ &:= \{c = (U, f, v, o, t, valid, \alpha) \in C \mid o = '=' \wedge pred(v, arg) = false\} \\ M_{true}^- &:= \{c = (U, f, v, o, t, valid, \alpha) \in C \mid o \neq '=' \wedge pred(v, arg) = true\} \\ M_{false}^- &:= \{c = (U, f, v, o, t, valid, \alpha) \in C \mid o \neq '=' \wedge pred(v, arg) = false\} \end{aligned}$$

Der Wahrheitswert der Auswertung $eval(pred, arg)$ eines Anfrageatoms ergibt sich dann für *einwertige Kontextmerkmale* als:

- wahr (*t*): $M_{true}^+ \neq \emptyset$ und $M_{false}^+ = M_{true}^- = \emptyset$, d.h. es existieren (allerdings mindestens 1) nur zutreffende Fakten.
- falsch (*f*): $M_{true}^+ = \emptyset$ und ($M_{false}^+ \neq \emptyset$ oder $M_{true}^- \neq \emptyset$), d.h. es existieren (allerdings mindestens 1) nur negative Fakten.

- unbekannt (u): $M_{true}^+ = M_{false}^+ = M_{true}^- = \emptyset$, d.h. es existieren keine Fakten
- widersprüchlich (i): $M_{true}^+ \neq \emptyset$ und $(M_{false}^+ \neq \emptyset$ oder $M_{true}^- \neq \emptyset)$, d.h. es existieren beide Fakten.

Bei *mehrwertigen* Attributen besitzen die Mengen M_{false}^+ und M_{false}^- keinen Aussagewert, da es keinen Widerspruch darstellt, dass das Merkmal auch andere Werte besitzt. Damit können die Auswertungsbedingungen vereinfacht werden zu

- wahr (t): $M_{true}^+ \neq \emptyset$ und $M_{true}^- = \emptyset$
- falsch (f): $M_{true}^+ = \emptyset$ und $M_{true}^- \neq \emptyset$
- unbekannt (u): $M_{true}^+ = M_{true}^- = \emptyset$
- widersprüchlich (i): $M_{true}^+ \neq \emptyset$ und $M_{true}^- \neq \emptyset$

In diesem Fall lässt sich auch »query shipping« zum Kontextfaktendienst für die *value-Condition* nutzen.

Für die Auswertung der komplexen booleschen Verknüpfungen bietet sich die Logik von BELNAP an ([Bel77], [Web98]), die durch folgende Wahrheitstabellen charakterisiert wird:

\neg	t	f	u	i			t	f	u	i			t	f	u	i
	f	t	i	u			t	t	f	u	i		t	t	t	t
	f	f	f	f			f	t	f	u	i		f	t	f	u
	u	u	f	f			u	t	u	u	t		u	t	u	t
	i	i	f	f			i	t	i	t	i		i	t	i	i

Abbildung 14.1.: Belnaps vierwertige Logik [Web98]

Als Schlussbemerkung sei angemerkt, dass dieser Anfragetyp sich auch so variieren lässt, dass auch explizit eine Auswertung nach der Konfliktauflösung gefordert werden kann (d.h. Auswertung findet nicht auf den Fakten statt, sondern auf der konfliktbereinigten Menge der Kontextinformationen). Dann reduziert sich die vierwertige Logik auf eine dreiwertige Logik (der Fall *widersprüchlich* tritt nicht mehr auf). Die Wahrheitstabellen bleiben analog gültig.

14.3.3. Anfragen nach Benutzern (*UserQuery*)

Der dritte Anfragetyp zielt nicht auf Kontextinformationen, sondern auf Benutzer ab, die bestimmte Kontextbedingungen erfüllen müssen. Ergebnismengen solcher Anfragen bestehen also aus Benutzern und haben die Struktur:

$getUsers(valid^*, T^*, \alpha^*, fvocond, mode)$

wobei *mode* kennzeichnet, wie bei der Auswertung der Anfragebedingung verfahren werden soll. Bei »STRICT« sollen widersprüchliche und unbekannte Ergebnisse zu *falsch* ausgewertet werden; bei »LOOSE« zu *wahr*.

```
SELECT | SUBSCRIBE    USERS
VALID                <dateRange>
ADDED                <dateRange>
WITH CONFIDENCE      >= <confidence>
WHERE                <featureValueCondition>
MODE                 STRICT|LOOSE
```

```
<featureValueCondition> -> <featureValueCondition> (AND|OR)
                           <featureValueCondition>
                           | NOT <featureValueCondition>
                           | <featureValueConditionAtom>
```

```
<featureValueConditionAtom> -> <featureURI> <predicate>
                               <valueExpression>*
```

Die Anfragebearbeitung ist hier mehrstufig:

- Die Anfrage $q = (ALL, F^*, valid^*, T^*, condition, \{=, \neq\}, \alpha^*)$ wird in einem ersten Schritt an den Kontextfaktendienst gestellt, wobei F^* die Menge aller in der *featureValueCondition* vorkommenden Kontextmerkmale ist.
- Das Ergebnis wird nach den in ihr vorkommenden Benutzern partitioniert. Für jeden Benutzer wird die Anfragebedingung geprüft, wobei die Auswertung analog zum vorangegangenen Abschnitt erfolgt.
- Ergibt die Auswertung der kompletten *featureValueCondition* u oder i , so wird der gesamte Ausdruck bei $mode = STRICT$ zu f , bei $LOOSE$ zu t ausgewertet.
- In die Ergebnismenge werden alle Benutzer, deren Bedingung zu t ausgewertet wurde, aufgenommen.

14.4. Kontinuierliche Anfragen

Wie auf der Kontextfaktenebene muss bei der Bearbeitung von Kontinuierlichen Anfragen das Ziel sein, eine Neuauswertung von Anfragen zu vermeiden. Durch die Möglichkeit,

auf Kontextfaktenebene kontinuierliche Anfragen zu registrieren, ist auf jeden Fall kein periodisches Polling erforderlich. Die Alterung kann durch das mitgelieferte »Ausfalldatum« ebenso leichter berücksichtigt werden. Standard-Strategie kann daher sein, dass bei Eintreffen eines neuen Faktums oder Wegfall eines Faktums durch Alterung die Anfrage neu berechnet wird. Dabei wird allerdings wieder die komplette Anfrage an den Kontextfaktendienst zur Auswertung geschickt und alle nachfolgenden Schritte aus der obigen Anfragebearbeitung erneut ausgeführt. Dies ließe sich vermeiden, wenn man das komplette Ergebnis des Faktendienstes zwischenspeichert; allerdings führt dies bei vielen Anfragen schnell zu Ressourcenengpässen, so dass dies auch keine Alternative darstellt.

In diesem Abschnitt soll daher untersucht werden, wie sich unter bestimmten Umständen die Neuauswertung von Anfragen bei neu eintreffenden Fakten vermeiden lässt. Dies ist besonders für *ContextQuery*s von Bedeutung. Für Anfragen nach Kontextinformationen über einen Benutzer ist für kontinuierliche Anfragen der Konfliktauflösungsoperator entscheidend. Hierbei spielt eine große Rolle, ob die Sicht, die durch den Konfliktauflösungsoperator berechnet wird, ohne erneute Ausführung der zugrundeliegenden Anfrage an den Faktendienst aktualisiert werden kann. Dies entspricht dem Konzept der *self-maintainability* von Sichten [GM95]. Erfüllt der Konfliktauflösungsoperator diese Eigenschaft, so kann einfach die Anfrage an den Faktendienst als kontinuierlich registriert werden. Bei neu eintreffenden Fakten muss der Konfliktauflösungsoperator nur entsprechend dem neuen Faktum die Sicht aktualisieren und bei Änderungen eine Benachrichtigung aussenden. Hierbei kann der Konfliktauflösungsoperator auf den zwischengespeicherten aktuell gültigen Wert plus ggf. Metadaten zurückgreifen.

Allerdings sind – im Gegensatz zu den üblich betrachteten Datenmodellen bei »continuous queries« – in unserem Fall negative Fakten zugelassen. Dies führt schon bei einfachen Anfragen zum Verlust der »self-maintainability«, da beim Eintreffen eines neuen negativen Faktums eine neuer Durchlauf der Konfliktauflösung mit allen Kandidaten erforderlich wird. In dieser Hinsicht werden sich Konfliktauflösungsoperatoren nicht unterscheiden.

Deshalb soll im folgenden untersucht werden, unter welchen Bedingungen und mit Hilfe von welchen Metadaten sich Neuberechnungen vermeiden lassen, zunächst im einfacheren Fall der einwertigen Merkmale:

- Bei *mostRecent* reicht es aus, den aktuellen Wert zusammen mit seinem Transaktionsdatum zu speichern. Ist der neu ankommende Wert ein positives Faktum und »jünger«, so wird der alte Wert ersetzt und eine Benachrichtigung ausgesandt. Ist der neu ankommende Wert ein negatives Faktum und entspricht sein Wert dem aktuell gültigen positiven Faktum, so ist allerdings eine Neuberechnung erforderlich. Ebenso ist eine Neuberechnung erforderlich, sobald das Auslaufdatum des aktuellen Wertes erreicht ist.

- Bei *mostNamed* wäre eine Statistik aller vorkommenden Werte vorzuhalten. Denn es reicht nicht aus, nur den aktuellen Wert mit seiner Häufigkeit zu speichern, sondern auch die Häufigkeiten der anderen Werte zusammen mit den individuellen Auslaufdaten, um bei Eintreffen neuer Fakten entscheiden zu können.
- *mostConfident* entspricht auf den ersten Blick strukturell *mostRecent*; statt des Transaktionsdatums ist die Konfidenz zu speichern. Dies gilt allerdings nur, wenn innerhalb der Anfrage Kontextmerkmale auf unterschiedlichen Abstraktionsebenen auch gleiche Alterungsfunktionen haben.
- Bei *mostSpecific* kann beim Eintreffen neuer Fakten überprüft werden, ob diese spezifischer sind. Neuberechnungen sind bei negativen Fakten und bei Erreichen des Auslaufdatums notwendig.

	Metadaten pro Anfrage	Neuberechnung erforderlich, wenn
CR	Transaktionsdatum	negatives Faktum, Erreichen des Auslaufdatums
CC	Konfidenz	negatives Faktum, Erreichen des Auslaufdatums (falls Alterungsfunktionen in Anfrage gleich)
CN	-	immer
CSF	Merkmal	negatives Faktum, Erreichen des Auslaufdatums
CSV	Wert	negatives Faktum, Erreichen des Auslaufdatums
CVV	Gültigkeitszeitraum	negatives Faktum, Erreichen des Auslaufdatums
CVD	Gültigkeitszeitraum, Ablaufdatum	negatives Faktum, Erreichen des Auslaufdatums

Tabelle 14.2.: Einordnung der Konfliktauflösungsstrategien in das Algorithmenschema

14.5. Verteilung

Bislang sind wir implizit davon ausgegangen, dass Kontextfakten in einer einzigen zentralen Faktenbasis lagern. Dies ist zwar der naheliegendste Ansatz, ist jedoch auch in mehrfacher Hinsicht problematisch.

- Aus Datenschutzgründen will man besonders in Unternehmen keine zentrale Datenbasis mit Verhaltensdaten über den einzelnen Mitarbeiter. Diese könnte nicht nur für die vorgesehenen Zwecke der Lernunterstützung, sondern auch darüber hinaus für eine Überwachung der Mitarbeiter genutzt werden, ohne dass der einzelne Mitarbeiter auch nur eine Chance hätte, Zugriffe nachzuvollziehen oder auch zu blockieren.

- Für unterschiedliche Kontextmerkmale kann es sinnvoll sein, diese getrennt voneinander zu speichern, um z.B. eine Replikation aus anderen Systemen zu vermeiden, die diese Daten führen (z.B. HR-Systeme). Die Anbindung kann dann auch durch virtuelle Integration realisiert werden, bei der die Anfragen direkt in die externen Systeme übersetzt werden.

Durch die klare Trennung von Kontextinformations- und -faktenebene ist der Übergang zu einer verteilten Architektur relativ einfach zu vollziehen. Wird die Partitionierung der Fakten nach Personen oder nach Merkmalen vorgenommen, so kann eine eintreffende Anfrage mit minimalem Aufwand an mehrere Faktenbasen weitergeleitet werden und ihre Ergebnisse können anschließend zu einer einzigen Ergebnismenge verschmolzen werden; zudem besteht kein Konsistenzproblem, da Fakten nur an genau einer Stelle vorliegen. Die Konfliktauflösungsstrategien sind hiervon nicht betroffen. Einzige Voraussetzung ist, dass die jeweiligen Faktenbasen die im vorangegangenen Kapitel definierte Anfrage-schnittstelle unterstützen. Da diese sich allerdings in ihrer Mächtigkeit auf das Wesentliche beschränkt, hält sich dieser Aufwand in Grenzen.

Vorteil dieser Verteilung wäre beispielsweise für den Mitarbeiter, dass die Kontextsammlung immer lokal erfolgt und beispielsweise ContextQuery's komplett lokal bearbeitet werden können. Erst für die Auswertung von UserQuery's ist ein entfernter Zugriff notwendig. Die hier eingehenden Anfragen können nachvollzogen und ggf. auch protokolliert werden.

Im verteilten Fall kann das Blackboard-Prinzip auf zwei unterschiedliche Arten (oder durch eine Kombination der beiden) umgesetzt sein:

- Die verteilten Faktenbasen stellen jeweils ein eigenes Blackboard dar, an das entsprechende Anreicherungsagenten gekoppelt sein können. Sie arbeiten also jeweils auf der Basis der lokalen Daten. Falls die Anreicherungsagenten auch Informationen mit einbeziehen würden, die in anderen Faktenbasen liegen (z.B. als Ergebnis anderer Anreicherungsagenten), besteht keine Übereinstimmung mit dem zentralen Fall.
- An einer zentralen Stelle wird eine eigene Faktenbasis durch die zentral wirkenden Anreicherungsagenten befüllt. Diese Anreicherungsagenten werden durch Änderungen an den verteilten Faktenbasen aktiviert, schreiben aber ihr Ergebnis in die zentrale Faktenbasis. Dadurch ist auch eine Kombination der verteilt vorliegenden Kontextfakten möglich, so dass auch kein Unterschied zum rein zentralen Fall besteht.

14.6. Fazit

Mit den in diesem Kapitel erarbeiteten Konfliktauflösungsoperationen und Anfragebearbeitungsalgorithmen ist für kontextbewusste Dienste eine Infrastruktur geschaffen, die ih-

nen ein hinreichend verlässliches Abbild der Situation des Benutzers bereitstellt, ohne dass sie sich um das Sammeln, das Anreichern und die Konsistenzsicherung kümmern müssen. Durch die Berücksichtigung von Gültigkeitszeiträumen, (abnehmende) Konfidenzen und potentiell widersprüchliche positive und negative Fakten wird der Unvollkommenheit der gesammelten Informationen bewusst Rechnung getragen und im Rahmen von Heuristiken zur Konfliktauflösung genutzt, um den nutzenden ein möglichst vollständiges *und* korrektes Abbild der Situation des Mitarbeiters bereitzustellen.

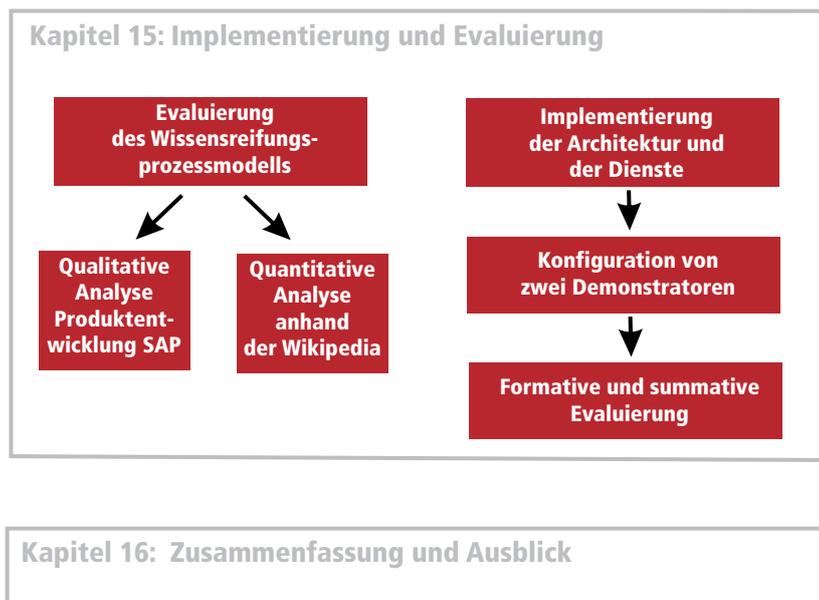
Teil V.

Evaluierung und Schluss

Problemstellung

Die vorgeschlagenen Modelle und Systemkonzepte müssen in realen Unternehmenssituationen umgesetzt und auf ihre Akzeptanz und Nützlichkeit hin überprüft werden.

Vorgehensweise



 **Implementierung/Evaluierung**

15.

Implementierung und Evaluierung von Architektur und Diensten

Ziel dieses Kapitel ist es, zu zeigen, dass (1) die vorgestellte Konzeptualisierung des arbeitsbegleitenden Lernens adäquat die Realwelt abbildet und (2) die in den vorangegangenen Kapiteln daraus entwickelte Unterstützung für das kontextgesteuerte Lernen in realen Unternehmensumgebungen umsetzbar ist und durch die Mitarbeiter akzeptiert und positiv bewertet wird. Im Sinne des integrativen Ansatzes der Arbeit geht es dabei nicht um die Evaluation von Einzellösungen in sehr stark dekontextualisierten, evtl. sogar vollautomatisierten Experimenten, sondern vielmehr um die Evaluation des Zusammenspiels mit realen Benutzern.

In einem ersten Schritt soll gezeigt werden, wie das in Teil II entwickelte Wissensreifungsmodell auf seine Validität hin überprüft wurde, bevor für die darauf aufbauenden Teile II, III und IV im Stile eines Design-Research-Ansatzes die technische Umsetzung kurz skizziert wird, die zur Grundlage für die zweistufige Evaluation mit Benutzern diente: eine formative Evaluierung und eine summative Evaluierung, beide im Kontext des Projektes *Learning in Process*.

15.1. Evaluierung des Wissensreifungsprozessmodells

Während die anderen Teile der Arbeit anhand einer Evaluation eines Testsystems überprüft werden können, ist dies für den Wissensreifungsprozess als theoretisches Konstrukt nicht ohne weiteres möglich. Deshalb wurde hier auf eine Evaluierung mittels einer qualitativen und einer quantitativen Studie gesetzt. Ziel dieser Evaluation ist es, die wesentlichen Aussagen der Wissensreifungstheorie zu überprüfen:

1. Die postulierten Phasen lassen sich tatsächlich in der Realwelt identifizieren, d.h. in

der Realwelt finden sich dem Wissensreifungsprozessmodell entsprechende Unterscheidungen und/oder es lassen sich charakteristische Übergänge anhand von Prozessen oder Instrumenten vorfinden.

2. Die postulierten Veränderungen der Eigenschaften zwischen den einzelnen Phasen lassen sich tatsächlich beobachten und eignen sich somit zur Beschreibung der unterschiedlichen Phasen.

Hierzu wurde zum einen der Entwicklungsprozess bei SAP im Rahmen einer Studienarbeit, zum anderen anhand der Artikelentstehungsprozess der offenen Wikipedia untersucht.

15.1.1. Qualitative Untersuchung

Im Rahmen der Studienarbeit von MARCUS ECHTER [Ech07], die vom Autor in Zusammenarbeit mit der Abteilung *Ramp-Up Knowledge Transfer (RKT)* bei SAP in Rot (inzwischen Waldorf) betreut wurde, wurde der Entwicklungsprozess bei SAP analysiert. Hierbei stand im besonderen Fokus die Entwicklung von Lernmaterialien (wie Online-Trainings und Schulungen) und ihre Verbindung zum Software-Entwicklungsprozess. Wie im Wissensreifungsprozessmodell postuliert, war auch bei SAP der Übergang von der Formalisierungsphase, in der Software-Dokumentation durch Entwickler erstellt wird, hin zur Ad-Hoc-Training-Phase mit Schulungen/Online-Kursen immer wieder ein großes Problem. Dabei war entweder die Qualität der Lernmaterialien oder ihr Fertigstellungszeitpunkt nicht ausreichend, so dass Konflikte mit der Auslieferung der Software auftraten. Ursache hierfür war auch die fehlende bzw. schlechte Kommunikation zwischen Fachexperten in den Entwicklungsabteilungen und den Trainings-Experten in der Abteilung, die für die Erstellung der Schulungsmaterialien zuständig war. So wurden meist die Dokumentationen nicht wiederverwendet, sondern vielmehr die Schulungsmaterialien komplett neu entwickelt.

Dies war eine der Motivationen, die sog. »Ramp-Up-Phase« einzuführen, die zwischen Entwicklung (*Develop*) und dem weltweiten Ausrollen der neuen Software-Version (*Deploy*) steht. Innerhalb dieser Phase wird das System bei ausgewählten Kunden installiert. Auch hierfür sind Schulungsmaterialien notwendig; allerdings sind diese näher an der Entwicklungsdokumentation und didaktisch noch nicht so ausgereift. Die Erstellung dieser Lernmaterialien (sog. »learning maps«) wird von der Abteilung *Ramp-up Knowledge Transfer (RKT)* koordiniert; die eigentlichen Inhalte werden allerdings von den Entwicklungsabteilungen geliefert. RKT berät nur bei der Aufbereitung und Auswahl. Die im Rahmen des Ramp-Up entstehenden Lerninhalte werden dann an die für die Lerninhalte und Schulungen zuständigen Abteilungen als Ausgangspunkt übergeben. Da sie bereits didak-

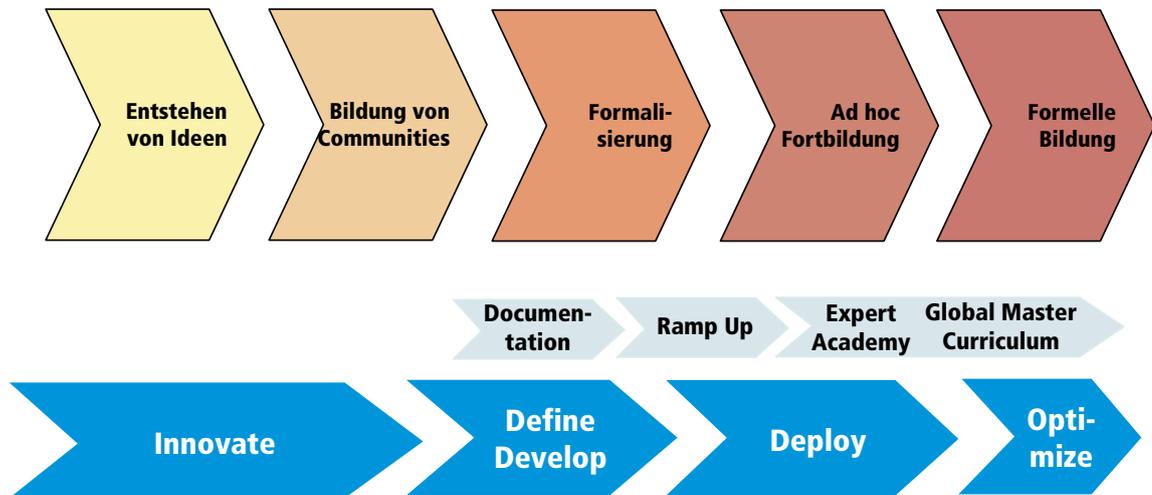


Abbildung 15.2.: Gegenüberstellung Wissensreifungsprozess und Produktinnovationsprozess von SAP

- Die organisatorische Trennung zwischen der Entwicklung und der Trainingsabteilung wird aufgeweicht, indem die Entwicklung in die Erstellung der »Learning Maps« einbezogen wird, die RKT-Abteilung allerdings eine gewisse didaktische Grundstruktur und -qualität sicherstellt.
- Die Überlappung von Entwicklungs- und Auslieferungsphase entspricht einer Verbesserung des Übergangs von Formalisierungs- zu Ad-Hoc-Phase (vgl. Abb. 15.2 für eine Gegenüberstellung mit dem Wissensreifungsprozess).
- Auch zeigte sich anhand des Fallbeispiels die Notwendigkeit der Ad-Hoc-Trainingsphase, bevor die Weiterbildungsprogramme mit Zertifizierung entwickelt werden können (entsprechen der formellen Weiterbildung).

15.1.2. Untersuchung anhand der Wikipedia

Ziel der Wikipedia-Untersuchung war eine quantitative Überprüfung, ob sich die postulierten Eigenschaften der einzelnen Phasen in Kapitel 3 tatsächlich beobachten lassen, d.h. ob sich entlang des Reifungsprozesses beispielsweise der explizite Vernetzungsgrad erhöht oder die Vermittelbarkeit steigt.

Um diese Untersuchung durchführen zu können, ist ein klarer Bezug zwischen Wissensreifungsprozess und dem Entwicklungsprozess von Artikeln in der Wikipedia herzustellen. Hierzu wurde eine qualitative Voruntersuchung durchgeführt, in der die unterschied-

lichen Instrumente und sozialen Prozesse analysiert und in Beziehung zum Wissensreifungsprozess gebracht wurden. Dabei hat sich u.a. ergeben [BS07], dass die Auszeichnungen für Artikel (lesenswert und exzellent) eine geeignete Einschätzung der Reife durch die Wikipedia-Community darstellen. Anhand der Analyse der Bewertungsrichtlinien für ausgezeichnete und exzellente Artikel wurden diese gleichgesetzt mit den Reifestufen Ad-Hoc-Training bzw. Formelle Bildung. Normale Artikel wurden mit der Formalisierungsstufe und Stubs¹ mit der Konsolidierungsphase gleichgesetzt.

Neben der Identifikation der Phasen im Lebenszyklus der Wikipedia-Artikel müssen noch die zu messenden Messgrößen identifiziert werden. Hierzu wurden die in Kapitel 3 identifizierten Kriterien wie folgt operationalisiert:

- **Vermittelbarkeit.** Hierzu wurde der Lesbarkeitsindex von AMSTAD [Ams78], der Strukturierungsgrad als Wörter pro Überschrift sowie die Bilder pro Anzahl Wörter eingesetzt.
- **Vernetzungsgrad.** Um diesen zu messen wurde die absolute Anzahl (eindeutiger) Links und die relative Anzahl von Links pro Wörter gemessen.

Für die quantitative Untersuchung [BS07] wurde der deutsche Wikipedia-XML-Dump mit Historien vom 24. Januar 2007 zugrundegelegt. Die Untersuchung wurde in zwei Teilen durchgeführt. Für den ersten Teil der Untersuchung (»Snapshot-Analyse«) wurde die jeweils neuste Version eines Artikels untersucht. Dabei wurde eine Stichprobe von 68.854 Artikeln gezogen. Aus diesen wurden spezielle Seiten, die keine Artikel repräsentieren, wie z.B. Benutzerprofile, Diskussionsseiten und Listen herausgefiltert (insgesamt 19.245). Innerhalb dieser wurde dann anhand der vorhandenen Auszeichnungen eine Klassifikation in »lesenswerte« Artikel und »exzellente« Artikel vorgenommen.

Die Ergebnisse der Snapshot-Analyse sind in Abb. 15.3 wiedergegeben. Insgesamt ließ sich ein statistisch signifikanter Unterschied (mit vorgegebenem $p < 0.05$) zwischen den einzelnen Artikelklassen bzgl. der Kriterien identifizieren, wenn auch die Standardabweichung relativ hoch war.

Der zweite Teil der Studie bestand aus einer Längsstudie von 343 exzellenten Artikel über ihre komplette Historie. Zur Normalisierung der Lebenszeit der Artikel wurden die Anzahl der Versionen auf 100 normalisiert.² Abb. 15.4 zeigt beispielhaft das Ergebnis für das Maß Wörter pro Überschrift. Auf der x-Achse ist dabei die Zeit als Prozentsatz der Gesamtlebenszeit angegeben; auf der y-Achse die Anzahl Wörter pro Überschrift. Zur Information ist beidseitig der Mittelwertlinie die Standardabweichung aufgetragen. Die Längs-

¹Dabei handelt es sich um »Rumpfartikel«. Da in der deutschen Wikipedia diese Auszeichnung seit 2005 nicht mehr explizit existiert, wurden Artikel mit weniger als 200 Wörtern als »Stubs« behandelt.

²In einer kleinen Stichprobe konnten keinerlei Unterschiede zur (berechnungstechnisch) deutlich aufwändigeren Normalisierung nach den tatsächlichen Bearbeitungszeitpunkten festgestellt werden.

	Stub	Normal	Good	Featured
#articles	19852	28763	612	343
#words p. article	43 (4) 60	1196 (753) 1359	5386 (4580) 3428	6689 (5952) 3561
readability p. article	35 (41) 32	50 (53) 35	55 (55) 7	54 (54) 6
#headlines p. art.	0 (0) 1	7 (5) 8	21 (18) 13	24 (22) 14
#images p. article	0 (0) 0	2 (0) 6	8 (6) 10	11 (8) 13
#words p. headline	105 (96) 50	198 (157) 198	391 (229) 880	438 (249) 883
#words p. image	132 (138) 45	747 (493) 827	1289 (722) 1897	1181 (647) 1633
#(unique) internal links	6 (1) 9 6 (1) 9	75 (55) 71 82 (58) 87	212 (170) 144 238 (188) 171	240 (213) 137 272 (233) 169
#words p. int. link	5 (3) 5	16 (13) 17	29 (21) 25	31 (24) 26
link density (%)	31 (33) 16	9 (8) 4	5 (5) 2	4 (4) 2

(mean (median) σ)

Abbildung 15.3.: Ergebnis der Snapshot-Analyse der Wikipedia [Mittelwert (Median) Standardabweichung]

studie bestätigte das Ergebnis der Snapshot-Studie: im Mittel lässt sich klar die postulierte Entwicklung der Kriterien zeigen, wenn auch die Standardabweichung hoch ist.

Die Wikipedia-Analyse hat gezeigt, dass auf der Basis der Artefakte, die bei der Wissensreife entstehen, die Kriterien Vermittelbarkeit (operationalisiert durch Lesbarkeit und Strukturierungsgrad) und explizite Vernetzung (operationalisiert durch Linkhäufigkeit und -dichte) den postulierten Verlauf zeigen. Hierbei lag der Fokus auf den Phasen Konsolidierung in Communities, Formalisierung und Ad-Hoc-Training. Begleitend dazu konnten Instrumente zur Förderung der Übergänge beobachten werden, z.B. in Form von Kandidatenlisten, die die entsprechende Aufmerksamkeit auf zu verbessernde Artikel lenken.

b) Words per headline development

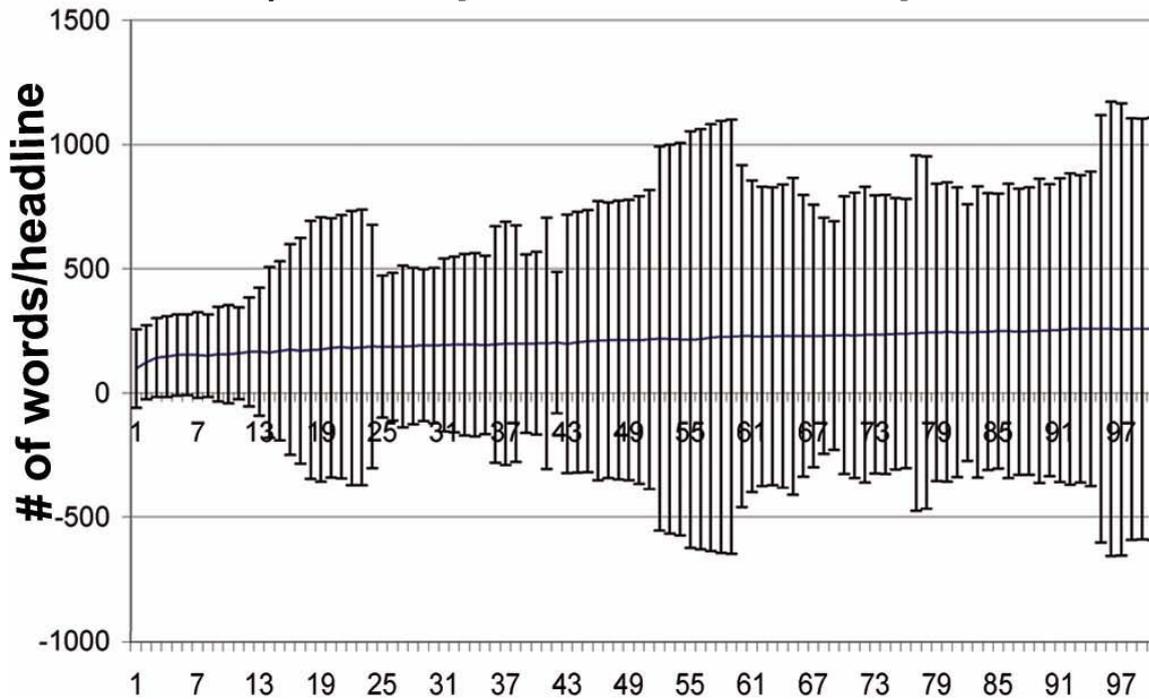


Abbildung 15.4.: Ergebnis der Längsstudie der Wikipedia anhand des Kriteriums Wörter pro Überschrift (Linie = Mittelwert, beidseitig ist die Standardabweichung aufgetragen, x-Achse gibt den Prozentsatz der Gesamtlebenszeit des Artikels an)

15.2. Implementierung

Mit der Implementierung der in den vorangegangenen Teilen erarbeiteten Konzepte soll gezeigt werden, dass sich die vorgestellten Lösungen auf der Basis aktueller Technologien tatsächlich umsetzen lassen und dass die so entstehende Lösung auch tatsächlich von den Benutzern als nützlich für ihre Lernprozesse wahrgenommen wird.

Die Umsetzung und Evaluierung erfolgte im Rahmen des von der Europäischen Kommission geförderten Verbundprojektes *Learning in Process*, das aus einem Konsortium von vier Forschungspartnern (FZI, INSEAD CALT, CIMNE und London Metropolitan University (LTRI)) und vier Industriepartnern (META4, CAS, Neurosoft, IBIT) bestand.

15.2.1. Überblick und Architektur des Prototypen

Der Prototyp wurde auf der Basis der vorgestellten dienstorientierten Architektur in Zusammenarbeit mit den Technologiepartnern³ umgesetzt wie in Abb. 15.5 dargestellt. Im folgenden sollen die einzelnen Komponenten näher vorgestellt werden

15.2.2. Endbenutzerinteraktion

Auf der Ebene der Endbenutzerinteraktion wurden die folgenden Werkzeuge realisiert:

- **SCORM-Laufzeitumgebung.** An der Referenzimplementierung *ADL SCORM RTE*⁴, die als servletbasierte Java-Anwendung realisiert ist, wurde die in Abschnitt 11.1 dargestellten Erweiterungen vorgenommen, um Lernobjekten Kontextinformationen zur Verfügung zu stellen. Hierzu wurde die darin enthaltene Implementierung des SCORM-API-Applets so erweitert, dass sie die CMI-Erweiterung unterstützt. Weitere Anpassungen der Oberfläche wurden durch INSEAD vorgenommen.
- **Lernassistent.** Die Funktionalität des Lernassistenten wurde auf drei verschiedene Varianten realisiert:
 - META4 hat auf der Basis des Microsoft Agent Frameworks⁵, das die einfache Erstellung von animierten Agenten mit Sprachausgabe ermöglicht, einen Lernassistenten entwickelt, der – ähnlich wie die Assistenten von Microsoft Office – erscheint, sobald eine Empfehlung vorliegt (vgl. Abb. 15.6).
 - Als zweite Realisierungsform wurde eine Desktop-Anwendung (in Java mit Desktopintegration) erstellt, die im Traybereich über ein Icon signalisieren kann, ob Empfehlungen vorliegen. Bei Bedarf kann ein Popup wie in Abb. 15.7 eingeblendet werden, das die Empfehlungen anzeigt.
 - Die dritte Realisierungsform wurde durch die CAS erstellt. Sie stellt eine Integration in das Intranetportal-Framework *CAS teamworks*⁶ dar, das auf der Basis des Microsoft IIS realisiert ist (vgl. Abb. 15.8). Dies wird dadurch erreicht, das bei vorliegenden Empfehlungen in die Portalausgabe eine kleine Empfehlungsbox eingebunden wird.
- **Kommunikation.** Für die Kontaktaufnahme mit anderen wurde zum einen durch INSEAD ihre Lotus Sametime-basierte Plattform *ICDT [Ang03]* erweitert und angepasst, die eine Kollaborationsinfrastruktur bereitstellt. Zum anderen wurde als

³Wo die Implementierung durch andere durchgeführt wurde, wird dies entsprechend in der Beschreibung gekennzeichnet.

⁴<http://www.adlnet.gov/scorm/20043ED/SRTE.aspx>

⁵<http://www.microsoft.com/msagent>

⁶<http://www.cas.de/produkte/teamWorks>

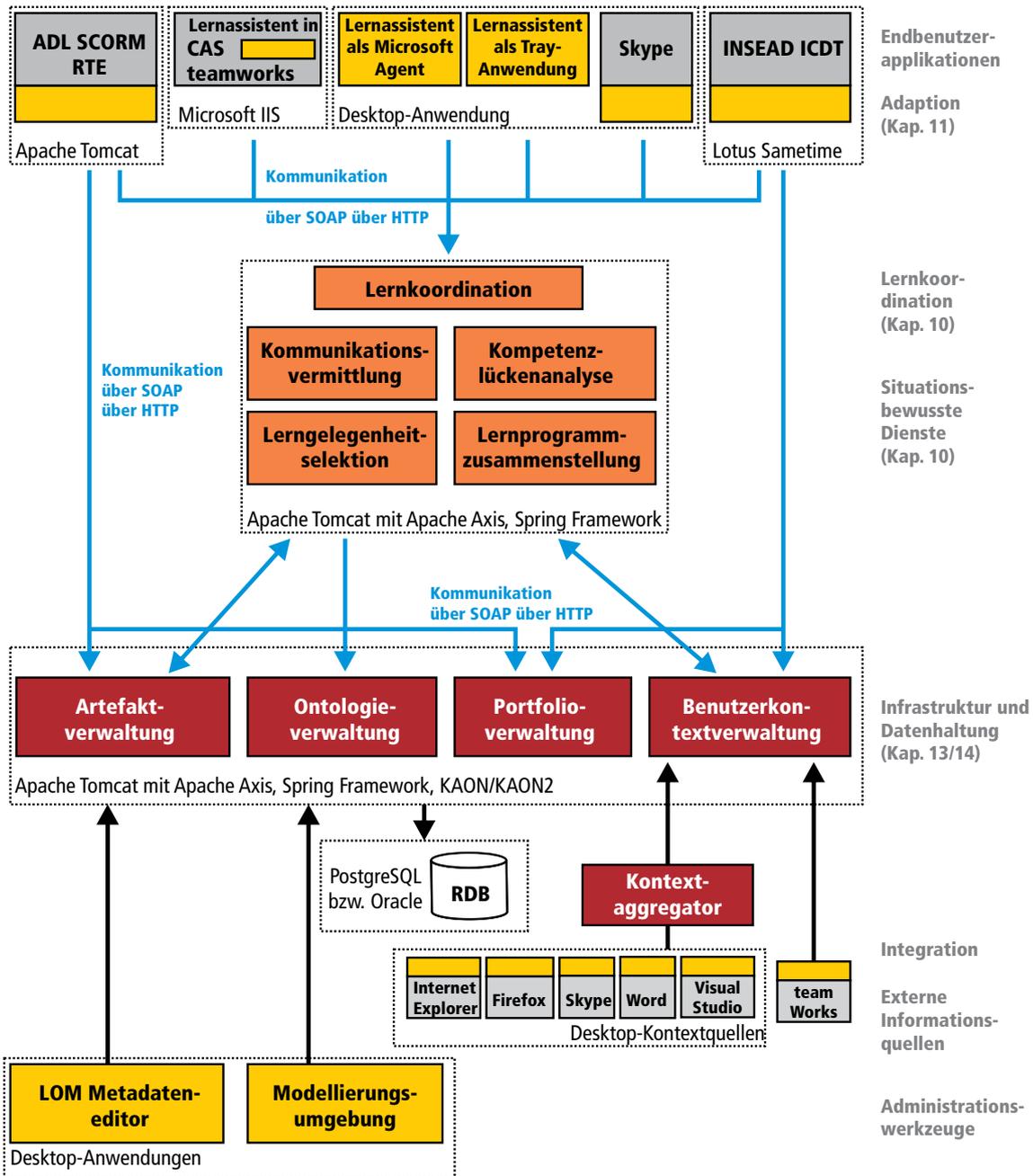


Abbildung 15.5.: Architektur des implementierten Gesamtsystems

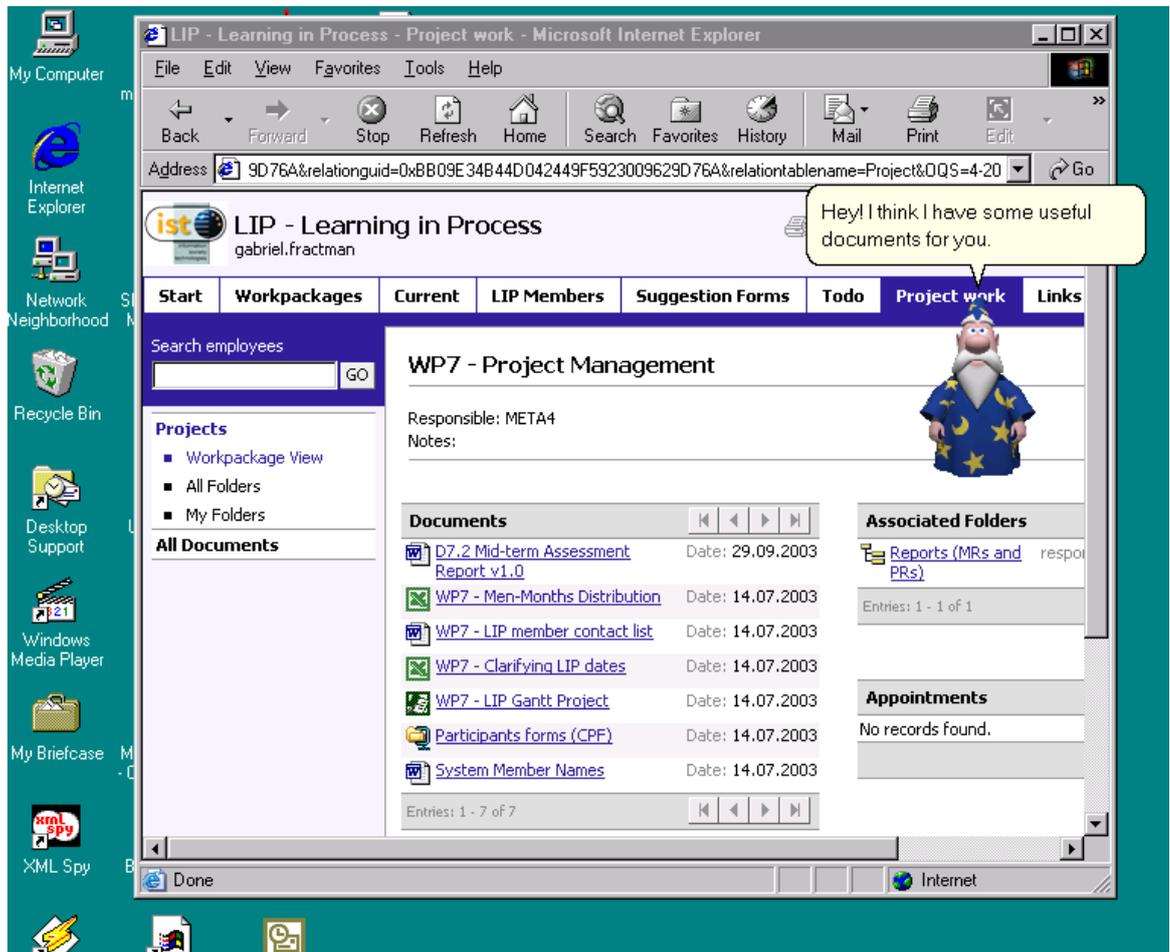


Abbildung 15.6.: Screenshot des Lernassistenten auf der Basis des Microsoft Agent Frameworks

leichtgewichtige Alternative der Skype-Messenger⁷ über die entsprechende API angebunden.

15.2.3. Lernkoordination und Situationsbewusste Dienste

Die Architekturebenen Lernkoordination und Lernunterstützende situationsbewusste Mehrwertdienste wurden auf der Basis von Apache Axis⁸ innerhalb von Apache Tomcat⁹ als Web Services implementiert. Als Framework für die komponentenorientierte Program-

⁷<http://www.skype.com>

⁸<http://ws.apache.org/axis/>

⁹<http://tomcat.apache.org/>

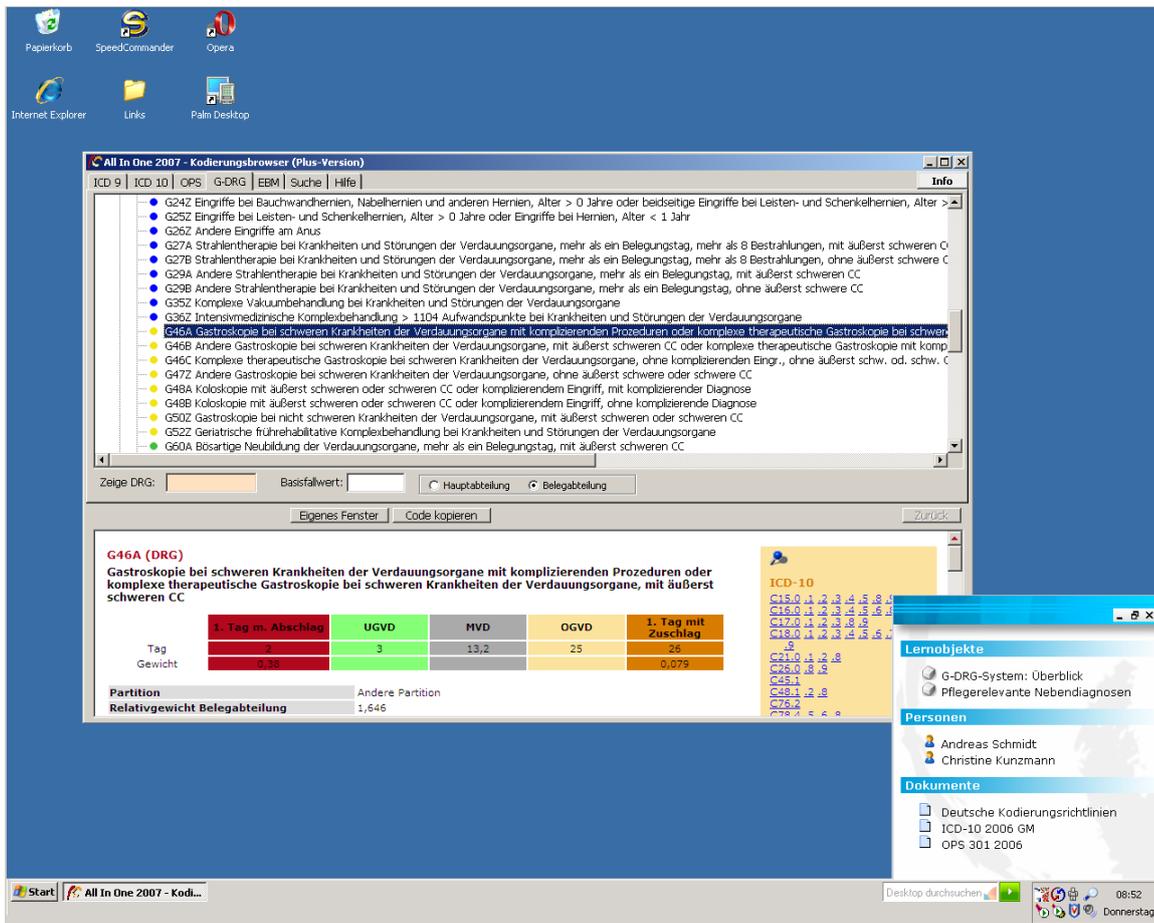


Abbildung 15.7.: Screenshot des Lernassistenten im Tray-Bereich

mierung mit sog. »dependency injection«, bei der die Verknüpfung der Komponenten und die Konfiguration zur Laufzeit mittels XML-Dateien erfolgt, wurde das Spring Framework¹⁰ eingesetzt. Dadurch war es auch möglich, die Einzeldienste sehr einfach sowohl mittels lokalen Aufrufen (für die Kommunikation zwischen den Diensten) als auch mittels Web-Service-Aufrufen bereitzustellen.

15.2.4. Infrastruktur und Datenhaltung

Für diese Ebene wurde auf dieselbe technische Infrastruktur (Axis, Tomcat, Spring) gesetzt. Für den Umgang mit Ontologien wurde auf KAON [MMS03] und KAON2 [Mot06] zurückgegriffen, die mit Hilfe eines Datalog-basierten Ansatzes in der Lage sind, auch

¹⁰<http://www.springframework.org>

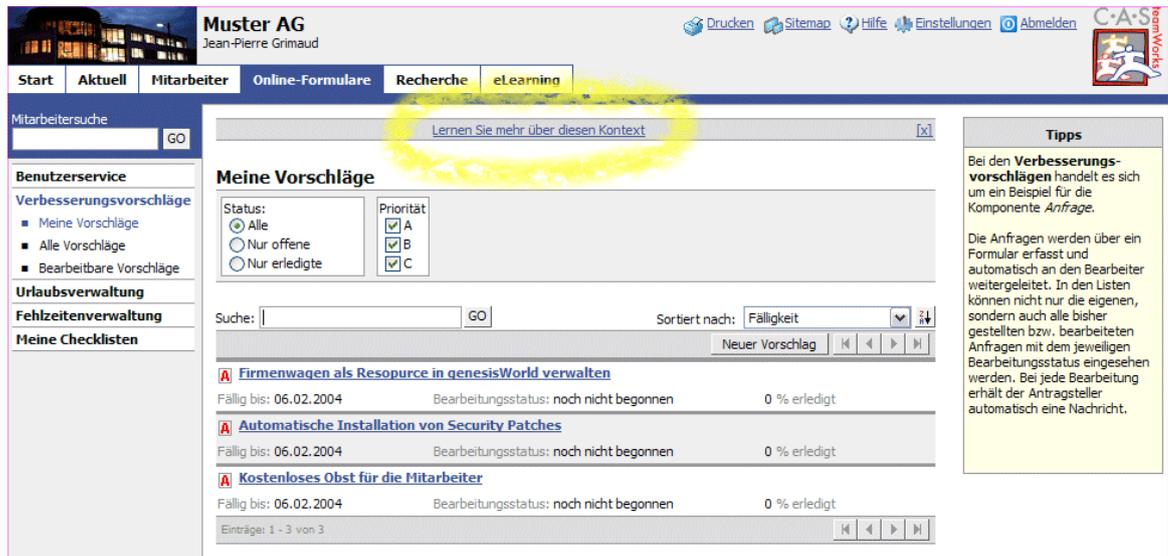


Abbildung 15.8.: Screenshot des Lernassistenten integriert in CAS teamWorks (Empfehlung hervorgehoben)

große Ontologien zu unterstützen, die in relationalen Datenbanken gespeichert sind. Als relationale Datenbank kam zum einen PostgreSQL in der Version 7, zum anderen Oracle 8i zum Einsatz. Auf diese setzte auch der Kontextmanagementdienst auf, dessen UML-Diagramme in Anhang C zu finden sind.

15.2.5. Integration und Externe Informationsquellen

Als Kontextquellen wurden die folgenden angebunden

- Mit einem *Plugin für Internet Explorer* (auf der Basis der sog. »browser helper objects«, entwickelt durch META4) und einem *Plugin für Mozilla* wurden besuchte URLs dazu herangezogen, um bei Intranetangeboten aus der zugrundeliegenden Struktur auf Projekte oder Prozessschritte zu schließen. Hierzu wurde die URL-Struktur des jeweiligen Intranetangebots auf Ontologieelemente abgebildet.
- Mit einem *Plugin für Microsoft Office* wurden die zugrundeliegenden Vorlagen der Dokumente, die Dateinamen sowie angezeigten Inhalte für die Ermittlung von Prozessschritten bzw. Aufgaben und Projekten genutzt. Hierzu wurden Zuordnungen von Vorlagen zu Ontologieelementen sowie stichwortbasierte Abbildungen (bei Dateinamen auf Projekte) benutzt (entwickelt durch META4).
- Im Falle des Demonstrators der CAS wurden aus der Serveranwendung von CAS

teamworks aus den ausgewählten Bereichen der Intranet-Anwendung die entsprechenden Kontextinformationen ermittelt (entwickelt durch die CAS).

- Für Softwareentwickler wurde durch META4 ein *Plugin für Microsoft Visual Studio* entwickelt, das aus Kommentaren Stichwörter extrahiert, die dann mit Ontologeelementen in Beziehung gebracht werden.
- Mit Hilfe eines Plugins für *Microsoft Outlook* wurden aus den Kontakten und dem Terminkalender Informationen zu sozialen Beziehungen und zu Verfügbarkeit/Stresszustand ermittelt.

Insgesamt konnten so wie in Abb. 15.9 illustriert die Kontextmerkmale teilweise automatisch, teilweise durch manuelle Angabe durch den Benutzer im Rahmen der Evaluierung berücksichtigt werden.

15.2.6. Administrationswerkzeuge

Für die Pflege der Domänenontologie wurde durch den Autor eine Modellierungsumgebung entwickelt (Abb. 15.10), die auf dem KAON [MMS03] OIModeler basiert. Für die Modellierung des Kompetenzkataloges und der Anforderungsprofile wurde ein Editor realisiert, der in Abb. 15.11 wiedergegeben ist. Er erlaubt die Zuordnung von Kompetenzen zu Organisationsentitäten durch einfaches Drag-and-Drop. Analog hierzu sind auch Möglichkeiten der Zuordnung von Mitarbeitern zu Organisationseinheiten sowie die manuelle Erfassung von Mitarbeiterkompetenzen möglich.

Für die Beschreibung der Lernobjekte wurde (da existierende Werkzeuge keine kompetenzorientierten Beschreibungen unterstützen) eine rudimentäre Autorenumgebung durch NEUROSOFT erstellt, die den LOM-Standard sowie die Einbindung der Kompetenzen als Voraussetzungen und Lernziele ermöglicht. Dies geschah in der dafür vorgesehenen Sektion *Classification*; hierbei wurden Kompetenzen als Taxonomieelemente (*taxon*) verstanden und mit den entsprechenden Ausdrucksmitteln von LOM repräsentiert.

15.3. Konfiguration der Demonstratoren

Auf der Basis dieses wiederverwendbaren Kernes wurden folgende organisationsspezifische Anpassungen entwickelt:

- ein Demonstrator bei der CAS AG in Karlsruhe, aufbauend auf ihrem Intranet-Produkt teamWorks
- ein Demonstrator bei IBIT in Palma de Mallorca

- eine Studie in der Entwicklungsabteilung von META4 in Madrid

Bei der CAS wurde eine Integration in die Intranetlösung teamWorks vorgenommen, so dass sowohl die Kontextgewinnung als auch die Anzeige von Lernempfehlungen (Lernassistent) direkt in die Intranetumgebung eingebettet ist. Demgegenüber wurde bei IBIT auf einen eigenständigen Lernassistenten gesetzt, der auf zwei Arten implementiert wurde: (1) auf der Basis von Microsoft-Agent-Technologie in Zusammenarbeit des Autors mit META4, (2) als sog. Tray-Anwendung, die bei Bedarf eine Sidebar öffnet.

Diese beiden Demonstratoren dienten als Basis für die Evaluierung der Benutzerakzeptanz, die im folgenden Abschnitt beschrieben wird. Für die Tests bei IBIT und CAS wurden im Verlauf des Projektes jeweils Kompetenz-, Prozess- und Organisationsmodelle durch Vertreter der Unternehmen selbst erstellt¹¹ und entsprechende Benutzer angelegt. Als Lernmaterialien wurden bei IBIT vorhandene Dokumente genutzt, die »Best Practices« oder einfache Checklisten darstellen; in wenigen technischen Themen wurde auf allgemein verfügbare Kurse (z.B. von Sun über Java-Technologien) gesetzt (insgesamt rund 90 explizite Lernressourcen). Bei der CAS wurde neben firmeneigenen Texten im wesentlichen modularisierte Standard-Kurse, die von Microsoft bezogen wurden (insgesamt rund 100 Lernobjekte). Bei META4 wurden für die Studie, die weniger umfangreich als die anderen beiden Demonstratoren war, Überblicksdokumente (ca. 50) zu bestimmten Technologien eingesetzt, die in der Entwicklungsabteilung zum Einsatz kommen.

15.4. Evaluation

Ziel der Arbeit ist es, eine Verbesserung des arbeitsbegleitenden Lernens zu erreichen. Es geht also nicht um spezielle Einzelbeiträge, sondern um die Integration in eine Gesamtlösung, die eine angemessene konzeptionelle Unterfütterung hat. Deshalb ist das Hauptziel der Evaluierung, dieses Ziel zu überprüfen.

15.4.1. Vorüberlegungen

Im Evaluierungsmodell von SCHENKEL [TS04] (als Erweiterung des Modells von KIRKPATRICK [Kir94]) werden die unterschiedlichen Arten der Evaluation in Ebenen strukturiert: Auf der Ebene 0 findet sich die Produktebene. Evaluierungen auf dieser Ebene umfassen Usability-Studien. Auf der Ebene 1, der Reaktionsebene werden die unmittelbaren Reaktionen der Teilnehmer bzgl. Nützlichkeit und Akzeptanz gemessen. Ebene 2 geht einen Schritt weiter und versucht zu ermitteln, wie der Effekt auf das Lernen an sich ist. Auf Ebene 3 wird bewertet, inwieweit eine Bildungsmaßnahme tatsächlich auch zu (dauerhaften)

¹¹Der Aufwand hierfür betrug in etwa 1 Woche Arbeitszeit.

Verhaltensänderungen am Arbeitsplatz geführt hat. Auf der Erfolgsebene (Ebene 4) findet zusätzlich ein Abgleich mit den ursprünglichen Zielen im Geschäftskontext verglichen und auf Ebene 5 die Investition dem monetären Nutzen gegenübergestellt. Hierbei steigt der Aufwand für eine solide Evaluation mit den unterschiedlichen Ebenen sehr stark an, so dass – trotz der unbestrittenen Nützlichkeit – die meisten Evaluationen meist maximal Ebene 2 berücksichtigen. Hauptgrund hierfür ist, dass auf den höheren Ebenen größere Langzeitstudien mit Kontrollgruppen benötigt würden, die in realen Unternehmenskontexten nicht zur Verfügung stehen, so dass Effekte nicht eindeutig dem Evaluierungsgegenstand zugeordnet werden können.

Dieses Problem trifft auch auf die vorliegende Arbeit zu. Da es allerdings primär darum geht, der Situation der Mitarbeiter Rechnung zu tragen, die angemessene Lernformen in Ergänzung zu den existierenden formellen Lernformen wie Schulungen oder E-Learning-Kurse, ist der (schwierig zu führende) Nachweis einer Verbesserung des Lernens nicht unbedingt notwendig; vielmehr kommt es darauf an, ob es von den Betroffenen als nützlich wahrgenommen wird. Denn nur wenn diese Wahrnehmung gegeben ist, werden die entsprechenden Werkzeuge auch eingesetzt und kann die Lernunterstützung tatsächlich stattfinden.

Dieser auf Nutzerurteile sich abstützenden Sichtweise entsprechend wurde auch eine Evaluierungsmethodik gewählt, die sich in zwei Phasen gliedert: in der ersten Phase geht es um das mitgestaltende Überprüfen und Verbessern des konzeptuellen Modells des situationsgesteuerten Lernens (im Rahmen einer formativen Evaluierung), in einer zweiten Phase um das Überprüfen der Akzeptanz eines auf den Diensten basierenden Demonstrators. Dies wurde im Kontext des Forschungsprojektes *Learning in Process* (LIP) durchgeführt. Um gerade in der formativen Phase größtmögliche Objektivität und wenig Beeinflussung der Testpersonen zu erreichen, wurde die Evaluierung durch JOHN COOK und CLAIRE BRADLEY vom Learning Technology Research Institute der London Metropolitan University koordiniert und methodisch begleitet.

15.4.2. Phase 1 - Formative Evaluierung

Für die formative Evaluierung wurden Mitarbeiter der beiden Anwendungspartner (CAS AG in Karlsruhe und IBIT in Palma de Mallorca) mittels szenariobasierten Techniken befragt ([Car95], [ECTB97]), die rein papierbasiert ablaufen (also noch kein System existiert/demonstriert wird). Da LIP ein recht komplexes System darstellt, war es unerlässlich, eine Einführung in die Systemvision zu geben, bevor die Testpersonen mit der Bearbeitung der Aufgabe begonnen haben. Die Aufgabe für die Testpersonen war es, ein Szenario zu entwickeln, wie sie mit dem System interagieren würden und welche Einzelaktionen sie hierzu durchführen würden. Ein Beispiel für eine komplette Beschreibung solcher Benutzerszenarien findet sich in Anhang D; sie besteht aus einer Charakterisierung

des Arbeitskontextes, des Zieles der Systeminteraktion und den Aktionen zur Erreichung des Zieles.

In einem Nachbearbeitungsschritt wurden die Szenarien mit dem Ziel analysiert, gewünschte Systemelemente («envisioned artefacts») herauszuarbeiten. Die gewünschten Elemente wurden dann kategorisiert und dienten als Input für die weitere Konzeption und Implementierung. Die genaue Methodik hierzu sowie die detaillierten Ergebnisse sind in [CB03] und [CBF04a] beschrieben.

Die wesentlichen Ergebnisse der formativen Evaluierung waren:

- Obwohl die meisten Evaluierungsteilnehmer nur klassische Fortbildungen oder große E-Learning-Kurse kannten, sprachen sie auf das Grundkonzept sehr positiv an. Das Erzeugen von Szenarien zeigte, dass sie sich problemlos mit dem Konzept identifizieren und sie die vorgeschlagene Integration in die Arbeitsprozesse nachvollziehen können.
- Gegenüber der starken Orientierung an Artefakten im ersten Konzept stellte sich die Kommunikation mit Kollegen und Experten als ein wichtiges Element für die Testpersonen heraus. Neben reinen Inhalten ging es ihnen auch stark darum, die passenden Ansprechpartner bei weitergehenden Fragen zu erhalten und mit ihnen schnell in Kontakt zu treten. Hier wurden teilweise direkte Interaktionen per Instant-Messaging vorgeschlagen. Dies führte zur Erweiterung des Konzeptes um Personen, die empfohlen werden.

15.4.3. Phase 2: Summative Evaluierung

Hintergrund und Versuchsaufbau

Am Projektende wurde zusätzliche eine summative Evaluierung, getrennt nach Autorenwerkzeug und Bereitstellungsplattform, durchgeführt, wobei die Evaluierung hierbei nach unterschiedlichen Systemaspekten getrennt und anhand der jeweiligen unternehmensspezifischen Demonstratoren erfolgte. Bei den Anwendungspartnern CAS AG und IBIT wurden übergreifende Aspekte getestet, während in Experimentalsituationen bei INSEAD und META4 spezifische Fragestellungen überprüft wurden. Diese Aspekte sollen im folgenden kurz beschrieben werden; die ausführliche Dokumentation der Evaluierung (Versuchsaufbau sowie unaggregierte und interpretierte Ergebnisse) findet sich im Projektbericht D8.3 [CBF04b].

Im Rahmen der Evaluierung sollten folgende beiden Hypothesen getestet werden:

1. Die Bereitstellungsplattform ist in der Lage, Lernmaterialien bereitzustellen, die zur

Situation des Lernenden¹² passen.

2. Die unter Hypothese 1 formulierte Funktionalität wird so erbracht, dass sie für die Lernenden benutzbar ist.

Hypothese 1 wurde dabei getrennt nach den im Rahmen der Anforderungsermittlung definierten Lernformen kursgesteuertes Lernen, selbstgesteuertes Lernen (»personal steered«) und kontextgesteuert (entspricht dem Begriff des situationsgesteuerten Lernens der Arbeit) durchgeführt. Bei der CAS ($N = 11$) wurde das kursgesteuerte Lernen getestet; bei IBIT ($N = 10$) wurde das selbstgesteuerte Lernen getestet. Das für diese Arbeit besonders relevante »kontextgesteuerte Lernen« wurde sowohl bei der CAS als auch bei IBIT getestet.

Die auf Usability ausgerichtete Hypothese 2 wurde bei META4 ($N = 10$) (proaktive Bereitstellung) getestet; bei Testpersonen von INSEAD wurde schließlich auf der Basis des IBIT-Demonstrators eine »Perceived Ease of Use«-Untersuchung durchgeführt (vgl. Abb. 15.15).

Die Evaluierung wurde entsprechend üblichen Usability-Evaluierungen aufgabenorientiert durchgeführt: die Teilnehmer erhalten konkrete Aufgaben; deren Ausführung wird durch den Evaluierungsleiter beobachtet. Am Ende wird den Teilnehmern ein Fragebogen gegeben.

Ergebnisse

Trotz des beschränkten Angebotes an Lernmaterialien, das für die Evaluierung zur Verfügung stand, waren die Urteile der Benutzer der Benutzergruppen bei IBIT und CAS zum kontextgesteuerten Lernen grundsätzlich positiv (vgl. Abb. 15.13). Zwar gab es einen hohen Anteil an »neutralen« Bewertungen, der darauf zurückzuführen ist, dass im Rahmen des Projektes keine adäquaten Lernmaterialien erstellt werden konnten und das Angebot existierender Materialien nicht ausreichend war.¹³ Allerdings wurden die kontextbewussten Empfehlungen dort, wo Materialien bereitstanden, durchweg als positiv eingestuft.

Was die Benutzerbarkeit der proaktiven Systemfunktionalität (Lernassistent) anbelangt, so war das Ergebnis der Benutzerstudie bei META4 mit dem agentenbasierten Werkzeug uneingeschränkt positiv (vgl. Abb. 15.14).

Die gerade in einer Unternehmensumgebung potentiell problematische Tatsache, dass die Tätigkeiten des Mitarbeiters technisch überwacht und ausgewertet werden, um den Kontext zu ermitteln, wurde von den Teilnehmern überwiegend auch nicht als beunruhigend bewertet, auch ohne dass detaillierte Informationen gegeben wurden:

¹²Im Projektbericht und der zugrundeliegenden Evaluierung wurde von »work context« und »user profile« gesprochen, was aber semantisch äquivalent ist.

¹³Das ergab sich aus einer detaillierteren Untersuchung der Freitextantworten des Fragebogens.

- Bei der Evaluation des Lernassistenten waren nur 4 von 10 beunruhigt über das Loggen von Benutzeraktivitäten.
- Bei der Evaluation der anderen Systemaspekte (selbstgesteuertes bzw. kursgesteuertes Lernen), die ebenfalls durch Kontextinformationen besser unterstützt wurden, ergab sich, dass nur jeweils 2 von 10 (CAS, IBIT) wegen der Beobachtung beunruhigt waren.

Auch die Untersuchung des »perceived ease of use« ist wie in Abb. 15.15 überwiegend positiv.

15.5. Fazit

Im Rahmen der prototypischen Implementierung wurde gezeigt, dass die in den vorangegangenen Kapiteln vorgestellte Architektur umsetzbar ist. Die darauf aufsetzende Evaluierung (auf Reaktionsebene Evaluierungsmodell von KIRKPATRICK [Kir94]) ergab positive Hinweise, dass das situationsgesteuerte Lernen in der Praxis als Lernform generell positiv eingestuft wird.

Ein abschließendes Urteil würde einen breiteren Testrahmen (mehr Unternehmen, größere Benutzergruppen innerhalb der Unternehmen) erfordern. Eine Evaluierung auf der Lernebene (die die Lerneffektivität misst) wurde nicht durchgeführt, da sie überwiegend die didaktische Qualität der Lernressourcen misst und somit nicht aussagekräftig für den Ansatz ist. Die interessanten Ebenen »Verhalten« (Transfer des Gelernten in Verhalten am Arbeitsplatz) und »Ergebnisse« (führte Lernen zu einer Verbesserung aus betrieblicher Sicht) sind (sofern hierfür bereits geeignete Methoden existieren) nur durch aufwendige Langzeitstudien ermittelbar.

Es bleibt an dieser Stelle ebenso zukünftigen Arbeiten überlassen, das Gesamturteil über das System auf einzelne Teilaspekte herunterzubrechen und deren Effekt zu messen. Hierzu gehört beispielsweise die Kompetenzlückenberechnung in Abhängigkeit von unterschiedlich aufgebauten Kompetenzkatalogen, um so genauer zu ermitteln, welche Eigenschaften eine Kompetenzmodellierung erfüllen sollte, um optimale Ergebnisse bei der algorithmischen Berechnung zu erzielen.

Insgesamt steht und fällt der praktische Einsatz mit der Modellierung – das haben die Demonstratoren gezeigt. Während sich der Aufwand für die Modellierung durchaus im Rahmen hielt, stellte sich das »adäquate« Modellieren als kritisch heraus. Hierzu war eine Unterstützung durch den Autore erforderlich, um für das Systemverhalten wesentliche Kontextmodellierungsaspekte herauszuarbeiten, z.B. eine Abwägung der Modellierungstiefe bei Aufgaben/Prozessen und Kompetenzen in Abhängigkeit von den zu erwartenden Lerninhalten oder auch der Grundansatz zur Modellierung von Kompetenzen. Um

so eine Modellierung nachhaltig aktuell und nützlich zu erhalten, sollte auf eine iterative Modellierung gesetzt werden, die bedarfsorientiert weiterentwickelt wird, um so das Modellieren ohne praktischen Nutzen zu vermeiden und den Modellierungsaufwand zu begrenzen.

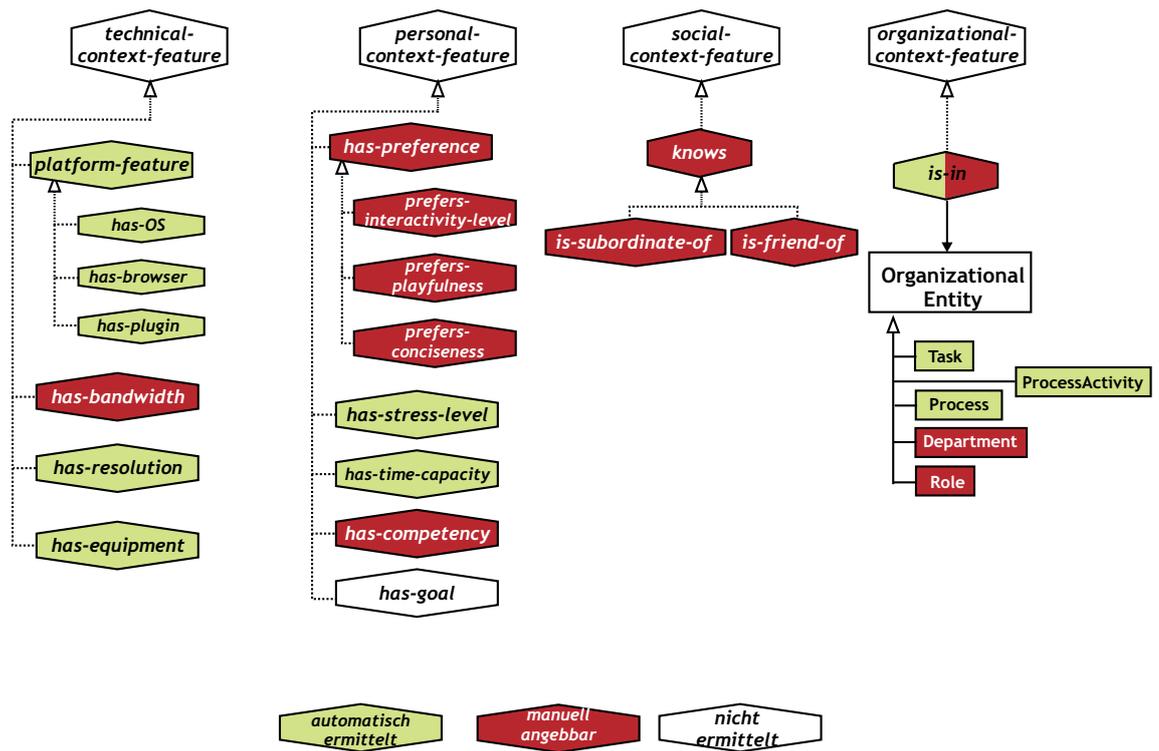
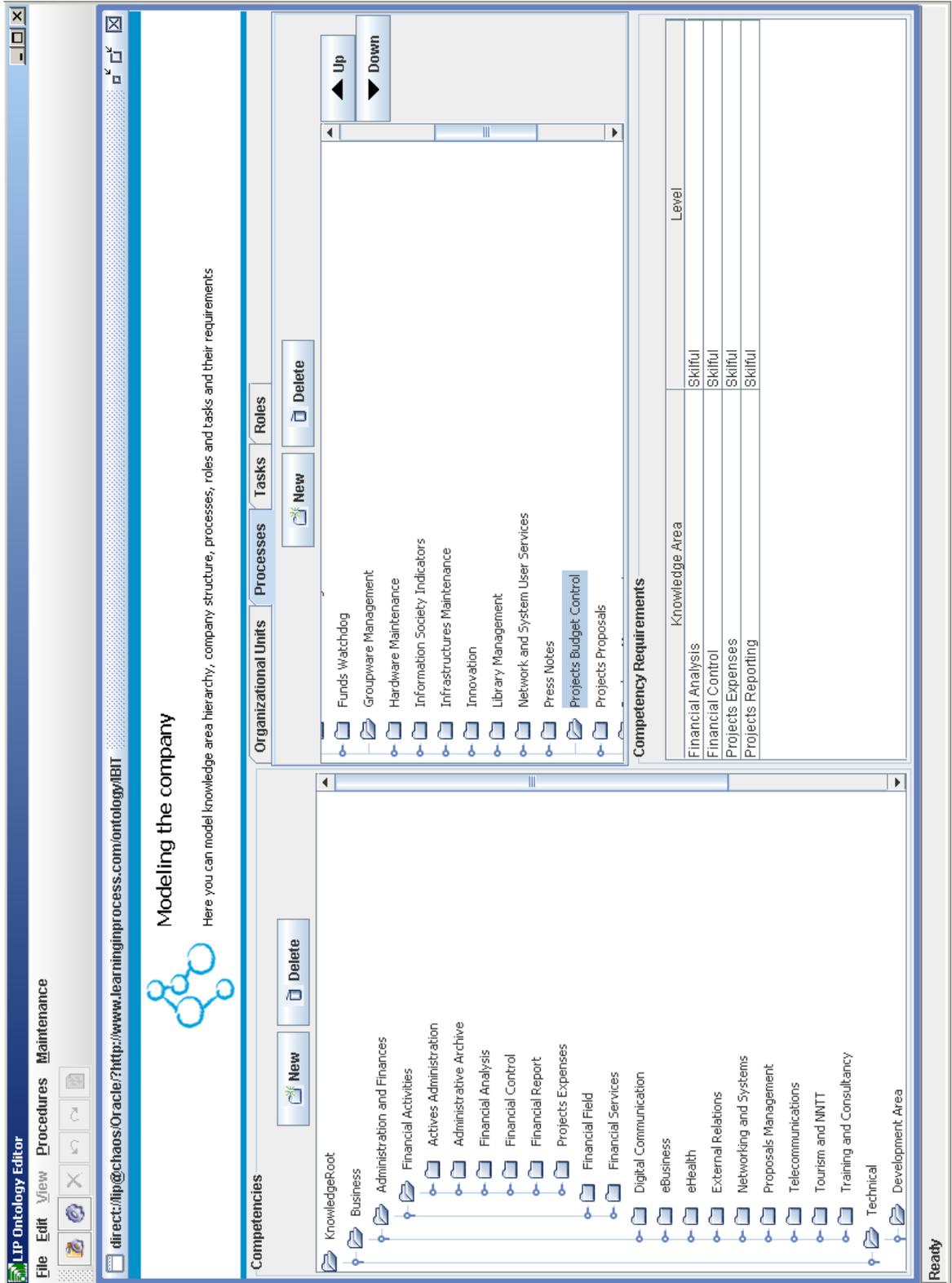


Abbildung 15.9.: Überblick über die berücksichtigten Kontextmerkmale im Rahmen der Evaluierung



Abbildung 15.10.: Screenshot Modellierungsumgebung



Anlage 11 Evaluierungsebenen

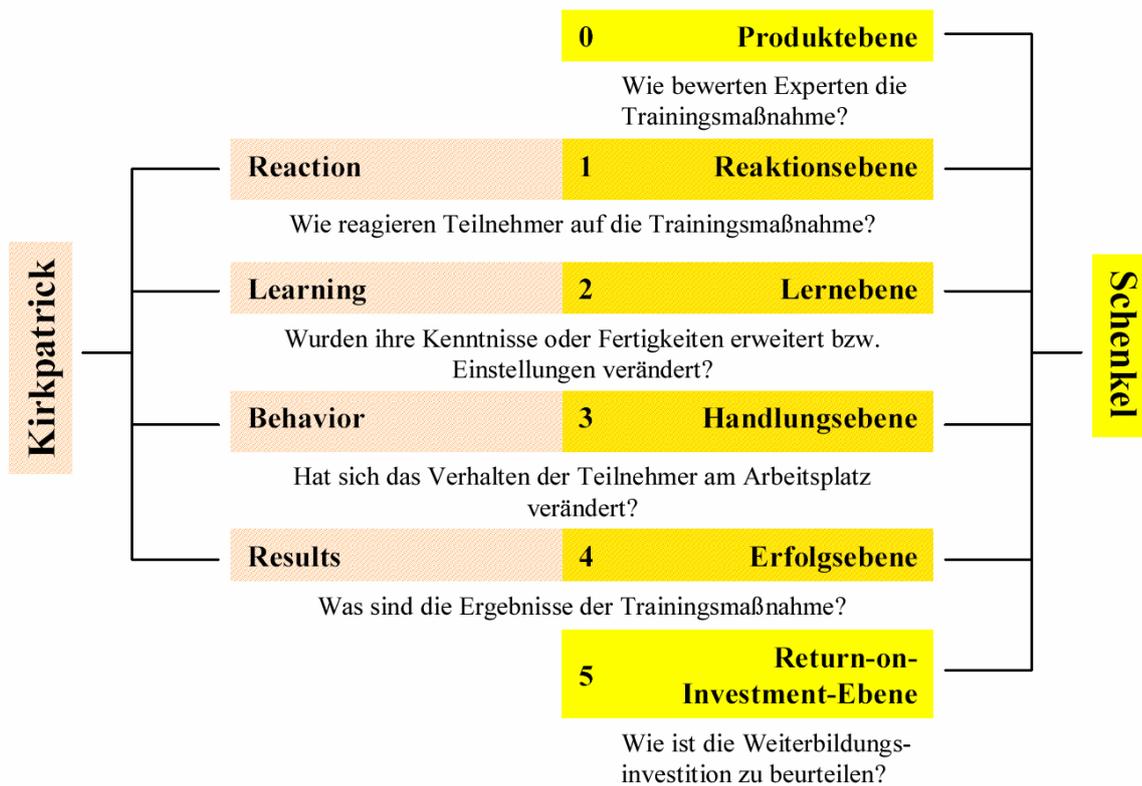
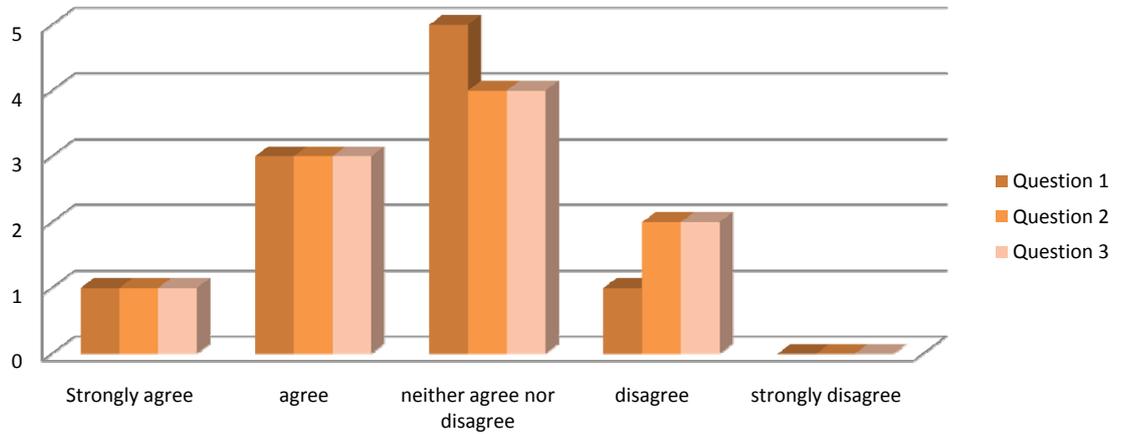


Abbildung 1: Evaluierungsmodelle nach Kirkpatrick und Schenkel⁷⁵
 Abbildung 15.12.: Evaluierungsebenen nach Schenkel und Kirkpatrick [TS04]

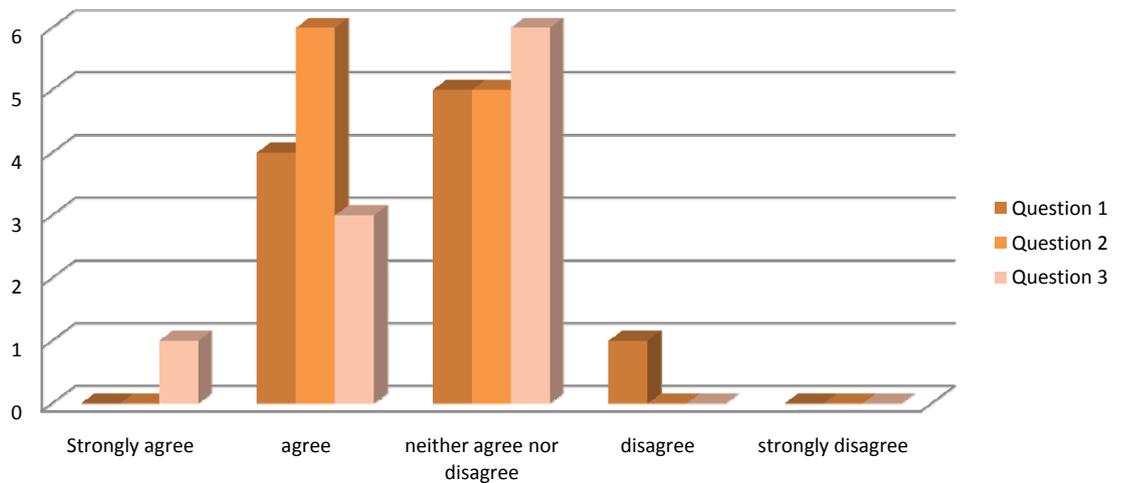
0. Produkt-ebene	1. Reaktions-ebene	2. Lernebene	3. Handlungs-ebene	4. Erfolgs-ebene	5. ROI-Ebene
<ul style="list-style-type: none"> ▪ Kriterienkataloge ▪ E-Learning Standards 	<ul style="list-style-type: none"> ▪ Digitalisierte Fragebögen ▪ Logfiles ▪ Interviews ▪ Kartenabfrage ▪ Beobachtung 	<ul style="list-style-type: none"> ▪ Multiple Choice ▪ Mehrfachauswahl ▪ Begriffsabfrage ▪ Lückentext ▪ Anordnungsaufgaben ▪ Zuordnungsaufgaben ▪ Freie Eingabe ▪ Präsenzprüfungen ▪ Logfiles 	<ul style="list-style-type: none"> ▪ Digitalisierte Fragebögen ▪ Arbeitsproben ▪ Wiederholung von Wissenstests ▪ Chats, Foren 	<ul style="list-style-type: none"> ▪ Digitalisierte Fragebögen ▪ Kennzahlen ▪ Vorher-Nachher-Messungen 	<ul style="list-style-type: none"> ▪ Vgl. Kosten E-Learning mit Kosten Präsenzseminar (Inputreduzierung) ▪ Vgl. ROI vor mit nach der Bildungsmaßnahme (Outputerhöhung)

Abbildung 2: Evaluierungsmethoden nach dem Model von Schenkel⁷⁶

Situationsgesteuertes Lernen (IBIT, N=10)



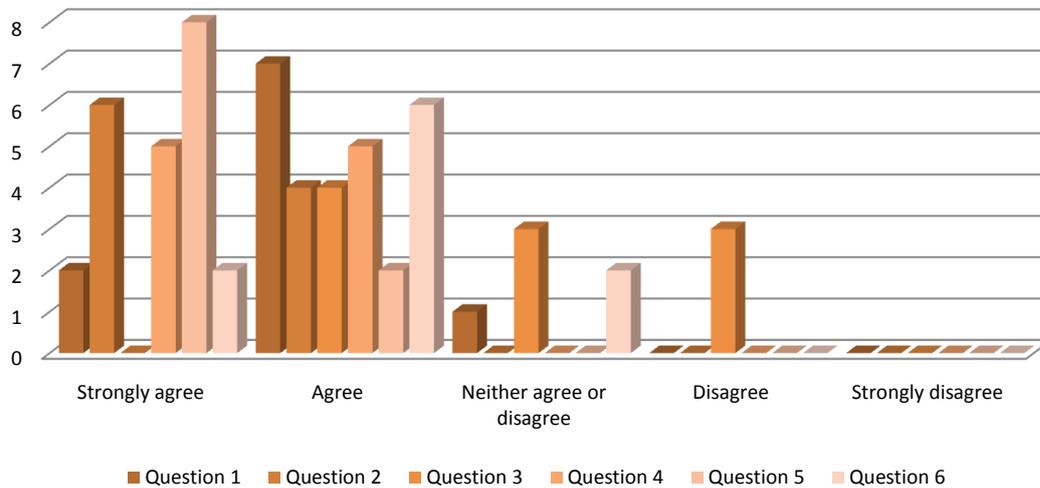
Situationsgesteuertes Lernen (CAS, N=10)



- Question 1** When I commence learning, the system suggests learning material relevant to my current situation.
Question 2 When I had a problem with what I was working, I was able to get satisfactory help from the system to resolve my problem.
Question 3 When I had finished my learning, taken a break and resumed learning, the system suggests only learning materials that I have not learned previously.

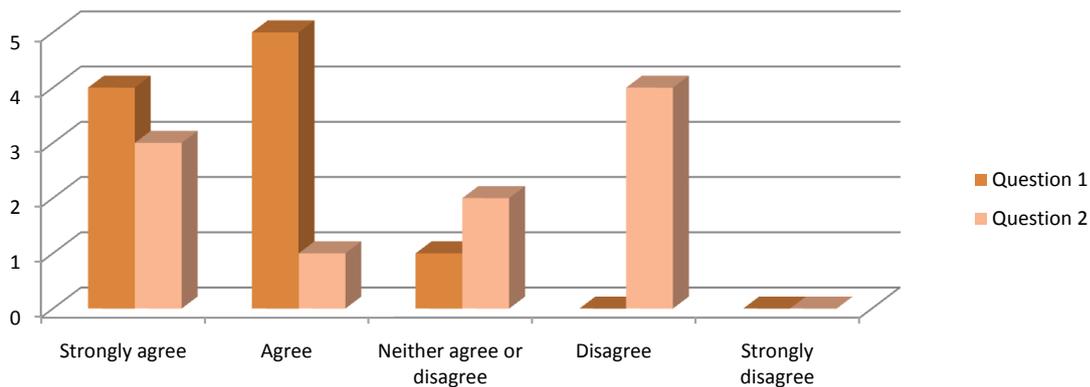
Abbildung 15.13.: Ergebnisse zum kontextgesteuerten Lernen (IBIT & CAS) [CBF04b]

Lernassistent zur Unterstützung des situationsgesteuerten Lernens (N=10)



- Question 1** It is possible to create a detailed profile of my work context and my learning needs, so that the system can recommend appropriate learning materials for me.
- Question 2** When I begin a new work context the system helps me by presenting learning materials that can help me.
- Question 3** When I have finished some learning materials, the system recommends further materials for me to continue my learning.
- Question 4** I am notified when new learning materials are created that match my profile.
- Question 5** After taking a break from learning, when I log back into the system it informs me of new Learning Objects that might be relevant for my work.
- Question 6** After taking a break from learning, when I log back into the system and continue a Learning Program Template, it resumes in the place where I left before.

Störung & Überwachung (N=10)



- Question 1** The system allows me to continue my work without much interference, only requesting my attention when necessary.
- Question 2** I do not mind the fact that the system is taking note of my activities (like opening a Word document or visiting a web site).

Abbildung 15.14.: Ergebnisse zum Lernassistenten (META4) [CBF04b]

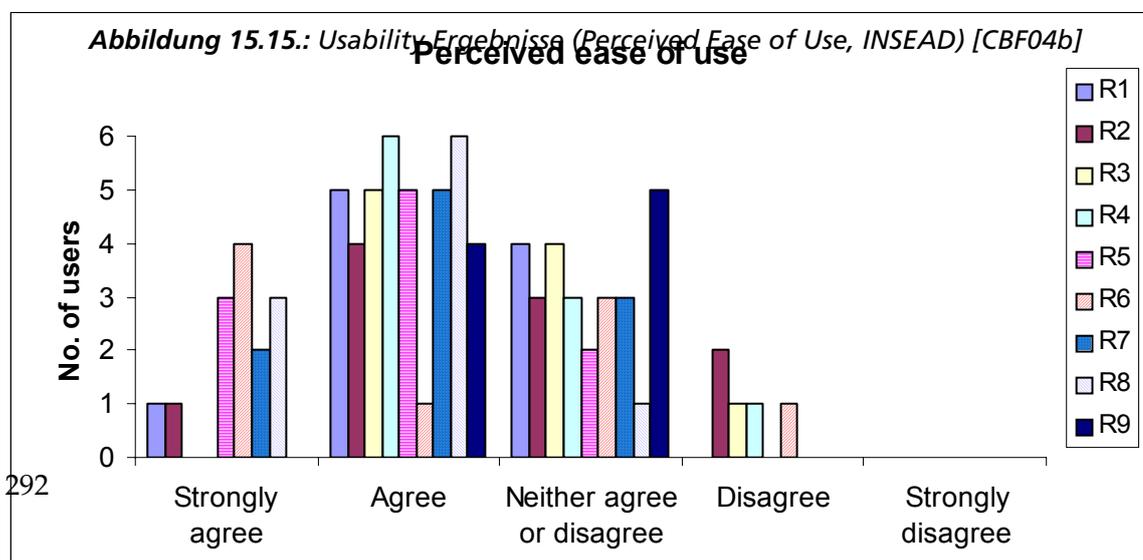
9. Evaluation Results and Analysis of Usability of LIP Delivery System

10 participants took part in this evaluation task carried out by INSEAD.

9.1 Summary of Results

Questionnaire: Perceived ease of use (INSEAD)

User requirements	Satisfaction ratings (total no. of participants)				
	5 Strongly agree	4 Agree	3 Neither agree or disagree	2 Disagree	1 Strongly disagree
1. LIP is easy to use.	1 5		4 0 0		
2. LIP is easy to learn to use.	1 4		3 2 0		
3. The system responded quickly to tasks I carried out.	0 5		4 1 0		
4. When I log into the system it takes me quickly to the relevant part of the system according to my user profile.	0 6		3 1 0		
5. I can easily post questions for experts or other learners to answer.	3 5		2 0 0		
6. I can quickly and easily give feedback on learning materials I have used.	4 1		3 1 0		
7. I can easily save files from the system to my PC.	2 5		3 0 0		
8. I can easily store learning materials for later use in my personal library.	3 6		1 0 0		
9. I would you use this system again.	0 4		5 0 0		



292

16.

Zusammenfassung und Ausblick

Im Rahmen der Arbeit wurde ein Konzept für das arbeitsbegleitende Lernen erarbeitet, umgesetzt und evaluiert, das aus einem konzeptuellen Makromodell (dem Wissensreifungsprozess), einer Methodik (dem situationsgesteuerten Lernen), einer Referenzontologie und -architektur sowie entsprechenden Diensten zur Umsetzung einer Lernunterstützung und einer adäquaten Kontextinfrastruktur besteht. Durch eine Evaluierung in realen Unternehmensumgebungen konnte gezeigt werden, dass die erarbeitete Lösung situationsbewusst Lerngelegenheiten empfiehlt und dies von den Benutzern akzeptiert wird.

Die Hauptbeiträge der Arbeit sollen im folgenden kurz charakterisiert und ihre Übertragbarkeit auf Problemstellungen jenseits derer dieser Arbeit dargelegt werden. Für etliche dieser Beiträge können auch konkrete Schritte und Entwicklungen skizziert werden, wie diese weiterverfolgt werden.

16.1. Wissensreifungsprozess

Der Wissensreifungsprozess liefert ein Modell, mit dem sich die komplexe Landschaft des betrieblichen Lernens und insbesondere des arbeitsbegleitenden Lernen strukturieren lässt. Es integriert sowohl den »Wissenstransfer« im Sinne des Wissensmanagements, als auch das »Lernen« im Sinne des geführten E-Learning.

Durch den Wissensreifungsprozess wird sichtbar, wo die Brüche zwischen unterschiedlichen Konzeptionen zu finden sind - und dass sie überwunden werden müssen. Das Modell liefert auch einen Ansatzpunkt, der die Hauptthese der Arbeit bildet: Durch die bessere systemseitige Berücksichtigung der Situation (»Situationsbewusstsein«) können Ansätze gefunden werden, die Brüche zu überwinden - oder zumindestens weniger gravierend zu gestalten. Dies kann geschehen

- durch die Verbindung der Inhaltserstellung für Lernmaterialien und früheren Phasen

des Prozesses, indem früher entstandene Artefakte (wie Notizen, Diskussionsbeiträge, Dokumente) sichtbar gemacht werden

- durch die Verbindung von informellen und formellen Lernschritten über gemeinsame Kontextelemente (z.B. im Rahmen von angepassten Learning Pathways)
- durch den Zugang zu Artefakten unterschiedlicher Reifestufen über einen einheitlichen, kontextunterstützten Zugang
- durch die Verbindung von Kompetenzen mit Arbeitskontexten (z.B. über Anforderungen und Ziele)

Das Modell des Wissensreifungsprozesses, das in dieser Arbeit zur Strukturierung der »Lernlandschaft« diente und somit eine konzeptionelle Verortung der Lernform des »situationsgesteuerten Lernens« ermöglichte, kann weit darüber hinaus Anwendung finden. In Zusammenarbeit mit MAIER wurde gezeigt [MS07], dass es das Potential eines Analysemodells für organisationale Lern- und Wissensaustauschprozesse hat und integrativ die unterschiedlichen Sichtweisen Wissensmanagement, E-Learning und Personalentwicklung bzw. Kompetenzmanagement zusammenführen kann. Über die reine Analyse hinaus kann das Modell auch zur Gestaltung von Lernunterstützung genutzt werden. Hierzu werden derzeit weitere und detailliertere empirische Untersuchungen (u.a. im Rahmen von ethnographischen Studien) im Rahmen des Integrierenden Projektes MATURE¹ durchgeführt, das durch die Europäische Kommission gefördert wird und in seiner Gesamtheit auf dem Wissensreifungsprozess basiert. Im Anschluss an die empirischen Untersuchungen sollen Werkzeuge und Dienste für die Überwindung der Hemmnisse und Barrieren in Reifungsprozessen konzipiert und implementiert werden.

Insbesondere lässt sich das universelle Reifungsprozessmodell auf spezielle Formen des Wissens spezialisieren und eröffnet hier neue Sichtweisen und neue Lösungsansätze. Dies gilt insbesondere für die Ontologieentwicklung (unter dem Stichwort »Ontologiereifung« bzw. »ontology maturing«). Hier wird in Zusammenarbeit mit BRAUN, WALTER und ZACHARIAS eine neue Methodik und dazu passende leichtgewichtige Werkzeuge für die inkrementelle, kollaborative und arbeitsintegrierte Formalisierung entwickelt (vgl. [BSWZ07], [BSW⁺07]), was in einer semantischen Social-Bookmarking-Anwendung (SOBOLEO) und einer kollaborativen semantischen Bildsuchmaschine ImageNotion erprobt wird. Angelehnt an das Wissensreifungsmodell lässt sich der Prozess der Ontologiereifung ebenfalls in Phasen strukturieren, zwischen denen charakteristische Übergänge zu finden sind.

Darüber hinaus wurden auch erste Schritte in Richtung einer Anwendung des Modells in Softwareentwicklungsprozessen unternommen. Zum einen wurde in Zusammenarbeit

¹<http://mature-ip.eu>

mit HAPPEL gezeigt, dass das Wissensreifungsmodell als Blaupause für die ein angemesseneres Modell für die Entwicklung von wiederverwendbaren Software-Komponenten dienen kann (vgl. [HS07]), wenn man den Grad der Wiederverwendbarkeit als Reifegrad und die entsprechenden Entwicklungsprozesse als Lernprozesse versteht. Auch sind an den Phasengrenzen Brüche zu überwinden, wie anhand von Fallstudien gezeigt wurde.

Darüber hinaus wird im Integrierten Projekt SOPRANO² im Bereich Ambient-Assisted Living eine neue Methode (»ontology centered design«) zur Systementwicklung eingesetzt, das Anforderungs- und Architekturentwurf bewusst als kollaborativen Lernprozess versteht, der durch eine »reifende« Ontologie als vermittelndes Artefakt unterstützt wird (vgl. [KSL07] und [BKSS08]). Die ersten Erfahrungen zeigen, dass diese Methodik gerade für geographisch verteilte Großprojekte Vorteile bietet und sich das Wissensreifungsmodell als Analyse- und Erklärungsmodell (und damit auch Instrument zur Vermittlung) sehr gut eignet.

16.2. Methodik des situationsgesteuerten Lernens

Die Methode des situationsgesteuerten Lernens kombiniert Ansätze aus dem Bereich der pädagogischen Agenten (proaktive Begleitung des Lernprozesses), des adaptive E-Learning und des kompetenzbasierten E-Learnings mit geschäftsprozessorientierter Informationsversorgung. Hierdurch können sowohl organisationale und persönliche sowie ergänzend soziale Aspekte Berücksichtigung finden. Im Gegensatz zu anderen kontext- oder situationsbewussten E-Learning-Methoden ist der Ansatz hierbei ganzheitlich und beschränkt sich nicht auf irgendwelche Teilaspekte.

Während bei der Vorstellung der ersten Entwürfe des situationsgesteuerten Lernens (z.B. [SW03]) durchaus große Überzeugungsarbeit geleistet werden musste, hat sich das Konzept unter Schlagworten wie »work-integrated learning« oder »embedded learning« durchgesetzt und wird beispielsweise im Rahmen des Integrierten Projektes APOSDLE [LM06] weiter verfolgt, wo stärker sog. »learning templates« (als instruktionelle Vorlagen) eingesetzt werden, um didaktische Grundstrukturen auch in von Mitarbeitern generierten Lernressourcen unterzubringen.

16.3. Referenzontologie und Referenzarchitektur für die arbeitsbegleitende Lernunterstützung

Mit der als »Professional Learning Ontology« bezeichneten Referenzontologie, die im Rahmen dieser Arbeit entwickelt wurde, wurde die Grundlage geschaffen, die Bereiche Wis-

²<http://www.soprano-ip.org>

sensmanagement, Personalentwicklung und E-Learning auch auf technischer Ebene zusammenzubringen. Gerade die Nutzung von Kompetenzen als Brücke zwischen organisationaler und individueller Perspektive hat sich bewährt, was auch die Erfahrung im Rahmen der Evaluierung zeigten.

Diese Konzeptualisierung hat Eingang gefunden in beginnende Standardisierungsbemühungen durch die *Ontology Outreach Advisory*³ (vgl. [JVM07]), die zukünftige HR-XML-Versionen um semantische Konzepte anreichern will. Die Auffassung von Themen oder Wissensgebieten als semantisch »abgeschwächte« Kompetenzen wird derzeit weiter in Richtung von kollaborativ, auf Tagging-Prinzipien aufsetzenden Methoden für das Kompetenzmanagement [BS08] ausgebaut.

Der Modellierungsrahmen bietet die Möglichkeit, zukünftig im Rahmen detailliertere empirischer Untersuchungen die Eignung von konkreten Kompetenzontologien zur Modellierung der Realität zu überprüfen, insbesondere im Bezug auf Modellierungsempfehlungen (Modellierungstiefe, Anzahl der Kompetenzrelationen etc).

Die dienstorientierte Referenzarchitektur definiert die Bausteine für eine auf diese Referenzontologie aufsetzende Infrastruktur und ihr Zusammenspiel. Durch die konsequente Dienstorientierung können flexibel unterschiedliche Teile genutzt werden und die Implementierungen flexibel ausgetauscht werden. Sie wurde beispielsweise auch als Grundlage für die im Auftrag von SAP untersuchte Integration in SAP-Produkte herangezogen und entsprechend konkretisiert.

16.4. Lernunterstützende Dienste

Mit den Lernunterstützenden Diensten wurden die Grundoperationen für das situationsgesteuerte Lernen umgesetzt: die Kompetenzenlückenberechnung, die kompetenzbasierte Lerngelegenheitsauswahl und die Zusammenstellung von Lernprogrammen. Im Gegensatz zu existierenden Ansätzen werden auch komplexere Kompetenzmodelle ausgenutzt, die Relationen wie Generalisierung zwischen Kompetenztypen und Komposition zwischen Kompetenzen unterstützen. Dadurch lässt sich auch die in der Praxis unvermeidbare Nutzung von Kompetenzen auf unterschiedlichen Abstraktionsniveaus und Granularitätsstufen unterstützen, indem eine Kompetenzsubsumtion in die Berechnung mit einfließt.

Diese Dienste sind dabei nicht auf das situationsgesteuerte Lernen beschränkt, sondern stellen Elementarbausteine für eine dienstorientierte Architektur für kompetenzorientierte Personalentwicklung [KS07] dar, wie sie von OSTYN als nächste Generation der HR-Infrastrukturen vorgestellt wird [Ost06].

³<http://www.ontology-advisory>

16.5. Kontextbewusste Lernobjekte

Das Konzept der kontextbewusste Lernobjekte zeigt einen einfachen Weg auf, wie man den Entwicklern von Lernobjekten nach dem SCORM-Standard zusätzlich Kontextinformation zur Verfügung stellen kann. Hierdurch werden neue Formen der Personalisierung ermöglicht, deren Effekt auf das Lernen noch gezeigt werden muss. Um zu zeigen, dass dieses Konzept auch kommerziellen Produkten zur Verfügung gestellt werden kann, wurde im Rahmen einer Diplomarbeit [Uhl06] das Konzept in das Futuna LMS integriert. Anhand eines Sprachlernkurses wurde exemplarisch demonstriert, wie sich beispielsweise anhand von affektiven Kontextfaktoren die Präsentation und die Auswahl von Aufgabentypen anpassen lässt. Hier sind nun Autoren von Lernobjekten gefordert, die Möglichkeiten didaktisch sinnvoll auszunutzen.

16.6. Kontextgewinnung und -verwaltung

Mit der erarbeiteten Kontextinfrastruktur wurde die Grundlage geschaffen, um das Konzept der Situation auch tatsächlich zu operationalisieren. Sie trägt der Herausforderung Rechnung, dass die Gewinnung von Kontextinformationen i.a. unsichere Ergebnisse liefert, und sich Kontextinformationen über die Zeit hinweg ändern, auch ohne dass das System das zwangsläufig mitbekommen muss. Der benutzte Formalismus sucht den Kompromiss zwischen semantisch reichhaltigen und ausdrucksächtigen Formalismen auf der Basis von Ontologien, die allerdings schlecht mit temporalen und unsicheren Daten zurecht kommen, und einfachen relationalen Ansätzen, für die effiziente Indexstrukturen existieren, mit denen sich die typischen Anfragen beantworten lassen.

Die Kontextinfrastruktur ist nicht auf den Anwendungsfall der Lernunterstützung beschränkt. Die Anforderungen einer skalierbaren Kontextverwaltung, die mit temporalen Aspekten und Unsicherheit umgehen kann und durch das Konzept der Alterung eine Balance zwischen Vollständigkeit und Korrektheit/Zuverlässigkeit erreicht, sind auch in anderen Bereichen relevant, etwa im Bereich der Ambient-Technologien und insbesondere im Ambient-Assisted Living (vgl. [KSHS08]). Dies zeigt sich beispielsweise im Projekt SOPRANO; hier wird der hier vorgestellte Ansatz als Grundlage für die Implementierung des Kontextmanagements eingesetzt, da sich gezeigt hat, dass beispielsweise auf OWL-DL aufsetzende Kontextmanagementansätze nicht angemessen für die Problemstellung sind.

Abbildungsverzeichnis

1.1. Gesamtüberblick über die Arbeit	9
2.1. Shneidermans Komponenten modernen Lernens [Shn02]	21
2.2. Konzeptuelle Darstellung von elektronischen Portfolios [TW04]	22
2.3. Zusammenhang von Unternehmensprozessen rund um das arbeitsbegleitende Lernen	25
2.4. Konstruktivistische Lernumgebungen nach Jonassen [Jon99]	31
2.5. Herleitung der Anforderungen aus den Thesen zum arbeitsbegleitenden Lernen	33
3.1. Gerichtete Wissenspyramide für die Charakterisierung von Kommunikationsprozessen in Anlehnung an [Nis02]	38
3.2. Nonakas SECI-Modell nach [NT95], eigene Darstellung	41
3.3. Wissensspirale nach Nonaka [Non94]	42
3.4. Nissens erweitertes Knowledge-Flow-Modell [Nis02]	43
3.5. Das Cynefin-Modell: Knowledge Flow	44
3.6. Modell für die organisationale Informationsverarbeitung nach Maier [Mai04] (eigene Darstellung aus [MS07])	46
3.7. Der Wissensreifungsprozess	47
3.8. Eigenschaften der einzelnen Phasen des Wissensreifungsprozesses	49
3.9. Ein Reifungsschritt im Detail	50
3.10. Situationsbewusste Unterstützung des Wissensreifungsprozesses	52
5.1. Informationsverhalten, Informationsrechercheverhalten und Informationssuchverhalten nach [Wil99]	74
5.2. Wilsons Kontext der Informationssuche [Wil81]	77
5.3. Niedzwiedzkas Modell des Informationsverhaltens[Nie03], aufbauend auf [WW96]	81
5.4. Informationsrecherche nach Choo [Cho02]	82
6.1. Situationssteuerung zwischen den Polen Fremd- und Selbststeuerung	94
6.2. Situationsgesteuertes Lernen: Prozessübersicht	99

6.3. Situationsgesteuertes Lernen im Kontext	104
7.1. Domänen- und Anwendungsentwicklungsprozesse, basierend auf [BKPS04]	115
7.2. Klassischer dienstorientierter Architekturstil vs. ontologiezentrierter dienstorientierter Architekturstil	116
8.1. Benutzungskontext und Benutzersituation	124
8.2. Benutzungssituation, Benutzermodell und Benutzerprofil	126
8.3. Konzepte der Aktivitätstheorie und die Zuordnung von Kontextkategorien nach [KO04]	131
8.4. Kontextmodell nach [KO04]	132
8.5. Identifizierte potentielle Kontextquellen zur Überprüfung der Ermittelbarkeit	134
8.6. Grundstruktur von LOM [Lea02]	135
8.7. Iterativer Abstimmungsprozess zur Erstellung der Kontextontologie im Rahmen des Projektes LIP	136
8.8. Erweiterte ER-Notation für die graphische Darstellung von Ontologien . . .	139
8.9. Kontextontologie: Überblick	140
8.10. Referenzontologie: Gesamtüberblick	143
8.11. Modellierung von Kompetenzen in der Referenzontologie (Detailansicht zu Abb. 8.10)	146
8.12. Referenzontologie: Modellierung der Anforderungsprofile (Detailansicht zu Abb. 8.10)	148
8.13. Das Domänenmodell und seine Integrationsfunktion mit den unterschiedlichen Gebieten des arbeitsbegleitenden Lernens	149
8.14. Vereinfachte Grundstruktur der Domänenontologie aus Abb. 8.10	150
9.1. Architekturmodell für kontextbewusste Anwendungen [HI06]	155
9.2. Referenzarchitektur für <i>Adaptive Educational Hypermedia Systems</i> [KS05b] . .	156
9.3. E-Learning Framework [WBR04]	158
9.4. Enterprise Knowledge Infrastructure [MHP05]	159
9.5. Infrastrukturdienstidentifikation anhand der Domänenontologie in Abb. 8.10	162
9.6. Situationsbewusste Dienste zur Berechnung von kontextabhängigen Relationen und Entitäten in Ergänzung zu Abb. 8.10	163
9.7. Referenzarchitektur lernunterstützender Systeme	164
9.8. Aufgaben der Schichten der Architektur und anzuwendenden Dekompositionsprinzipien	165
9.9. Rolle des Lernkoordinators: dynamische Sicht	171
9.10. Lösungsarchitektur	173

10.1. Überblick über den Prozess der lernunterstützenden situationsbewussten Dienste	177
10.2. Ausschnitt aus einer Kompetenzontologie für das Projektmanagement . . .	179
10.3. Ausschnitt aus einer Kompetenzontologie für das Projektmanagement (erweitert um Kompetenzbeziehungen)	183
10.4. Ausschnitt aus einer Kompetenzontologie für einen Software-Entwickler . .	184
10.5. Ausschnitt aus einer Kompetenzontologie für das Projektmanagement . . .	189
11.1. Grundkonzept eines webbasiertes Lernsystems [Ost05]	203
11.2. Struktur eines SCORM-Paketes [Adv04]	204
11.3. Klassische und kontextbewusste Lernobjekte aus SCORM-Sicht	206
11.4. Umsetzung kontextbewusster Lernobjekte mit SCORM-kompatiblen LMS .	207
11.5. Konzeptuelle Übersicht über die möglichen Realisierungsformen eines Lernassistenten	209
11.6. Interaktionsdiagramm zwischen Benachrichtigungskomponente und dem Lernkoordinator	211
11.7. Integration in Google Desktop (Haupt-XML-Datei)	212
12.1. CoBrA Überblick [Che04]	223
12.2. Architektur des Ansatzes von HECKMANN [Hec06b]	224
12.3. Architektur der Kontextverwaltung: Alternativen	230
12.4. Kontextverwaltungsinfrastrukturdienst: Konzeptionelle Architektur	232
13.1. Alternativen zur Kombination von Reasoning und RDBMS	245
14.1. Belnaps vierwertige Logik [Web98]	257
15.1. Product Innovation Lifecycle von SAP [Ech07]	269
15.2. Gegenüberstellung Wissensreifungsprozess und Produktinnovationsprozess von SAP	270
15.3. Ergebnis der Snapshot-Analyse der Wikipedia [Mittelwert (Median) Standardabweichung]	272
15.4. Ergebnis der Längsstudie der Wikipedia anhand des Kriteriums Wörter pro Überschrift (Linie = Mittelwert, beidseitig ist die Standardabweichung aufgetragen, x-Achse gibt den Prozentsatz der Gesamtlebenszeit des Artikels an)	273
15.5. Architektur des implementierten Gesamtsystems	275
15.6. Screenshot des Lernassistenten auf der Basis des Microsoft Agent Frameworks	276
15.7. Screenshot des Lernassistenten im Tray-Bereich	277

15.8. Screenshot des Lernassistenten integriert in CAS teamWorks (Empfehlung hervorgehoben)	278
15.9. Überblick über die berücksichtigten Kontextmerkmale im Rahmen der Evaluierung	286
15.10.Screenshot Modellierungsumgebung	287
15.11Screenshot Kompetenzmodellierung	288
15.12Evaluierungsebenen nach SCHENKEL und KIRKPATRICK [TS04]	289
15.13Ergebnisse zum kontextgesteuerten Lernen (IBIT & CAS) [CBF04b]	290
15.14Ergebnisse zum Lernassistenten (META4) [CBF04b]	291
15.15Usability Ergebnisse (Perceived Ease of Use, INSEAD) [CBF04b]	292

Anhang



Basisontologie in OWL

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"
  xmlns="http://www.fzi.de/ipe/LIP#"
  xmlns:swrl="http://www.w3.org/2003/11/swrl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xml:base="http://www.fzi.de/ipe/LIP">
  <owl:Ontology rdf:about="">
    <owl:imports rdf:resource="http://www.daml.org/rules/proposal/swrl.owl"/>
    <owl:imports rdf:resource="http://www.daml.org/rules/proposal/swrlb.owl"/>
  </owl:Ontology>
  <owl:Class rdf:ID="ContextEntity"/>
  <owl:Class rdf:ID="Flash">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="BrowserPlugin"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="LearningProgram">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#rest"/>
        <owl:allValuesFrom rdf:resource="#LearningProgram"/>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">
          >1</owl:cardinality>
        <owl:onProperty rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#rest"/>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:allValuesFrom>
          <owl:Class rdf:ID="LearningObject"/>
        </owl:allValuesFrom>
        <owl:onProperty rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#first"/>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Class rdf:ID="LearningResource"/>
    </rdfs:subClassOf>
    <rdfs:subClassOf rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#first"/>
```

```

        <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >1</owl:cardinality>
    </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Browser">
    <rdfs:subClassOf>
        <owl:Class rdf:ID="TechnicalEntity"/>
    </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="InternetExplorer">
    <rdfs:subClassOf rdf:resource="#Browser"/>
</owl:Class>
<owl:Class rdf:ID="Java">
    <rdfs:subClassOf>
        <owl:Class rdf:about="#BrowserPlugin"/>
    </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Mozilla">
    <rdfs:subClassOf rdf:resource="#Browser"/>
</owl:Class>
<owl:Class rdf:ID="Opera">
    <rdfs:subClassOf rdf:resource="#Browser"/>
</owl:Class>
<owl:Class rdf:ID="DreyfusLevels">
    <rdfs:subClassOf>
        <owl:Class rdf:ID="CompetencyLevel"/>
    </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Windows">
    <rdfs:subClassOf>
        <owl:Class rdf:ID="OperatingSystem"/>
    </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Competency"/>
<owl:Class rdf:about="#LearningObject">
    <rdfs:subClassOf rdf:resource="#LearningResource"/>
</owl:Class>
<owl:Class rdf:ID="Person">
    <rdfs:subClassOf rdf:resource="#LearningResource"/>
</owl:Class>
<owl:Class rdf:about="#OperatingSystem">
    <rdfs:subClassOf rdf:resource="#TechnicalEntity"/>
</owl:Class>
<owl:Class rdf:ID="OrganizationalContextEntity">
    <rdfs:subClassOf rdf:resource="#ContextEntity"/>
</owl:Class>
<owl:Class rdf:ID="MacOS">
    <rdfs:subClassOf rdf:resource="#OperatingSystem"/>
</owl:Class>
<owl:Class rdf:about="#BrowserPlugin">
    <rdfs:subClassOf rdf:resource="#TechnicalEntity"/>
</owl:Class>
<owl:Class rdf:ID="Linux">
    <rdfs:subClassOf rdf:resource="#OperatingSystem"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="has-loudspeaker">
    <rdfs:subPropertyOf>
        <owl:ObjectProperty rdf:ID="technical-context-feature"/>
    </rdfs:subPropertyOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="learning-object-property">
    <rdfs:subPropertyOf>
        <owl:ObjectProperty rdf:ID="resource-property"/>
    </rdfs:subPropertyOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#technical-context-feature">
    <rdfs:subPropertyOf>
        <owl:ObjectProperty rdf:ID="context-feature"/>
    </rdfs:subPropertyOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="has-bandwidth">
    <rdfs:subPropertyOf rdf:resource="#technical-context-feature"/>
</owl:ObjectProperty>

```

```

<owl:ObjectProperty rdf:ID="requires-competency">
  <rdfs:range rdf:resource="#Competency"/>
  <rdfs:domain rdf:resource="#ContextEntity"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="has-prerequisite">
  <rdfs:subPropertyOf rdf:resource="#learning-object-property"/>
  <rdfs:domain rdf:resource="#LearningObject"/>
  <rdfs:range rdf:resource="#Competency"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="prefers-interactivity">
  <rdfs:subPropertyOf>
    <owl:ObjectProperty rdf:ID="personal-preference"/>
  </rdfs:subPropertyOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="has-display">
  <rdfs:subPropertyOf rdf:resource="#technical-context-feature"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="social-context-feature">
  <rdfs:subPropertyOf>
    <owl:ObjectProperty rdf:about="#context-feature"/>
  </rdfs:subPropertyOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="prefers-conciseness">
  <rdfs:subPropertyOf>
    <owl:ObjectProperty rdf:about="#personal-preference"/>
  </rdfs:subPropertyOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#context-feature">
  <rdfs:domain rdf:resource="#Person"/>
  <rdfs:subPropertyOf>
    <owl:ObjectProperty rdf:about="#resource-property"/>
  </rdfs:subPropertyOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="consists-of-objects">
  <rdfs:range rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="organizational-context-feature">
  <rdfs:subPropertyOf rdf:resource="#context-feature"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#resource-property">
  <rdfs:domain rdf:resource="#LearningResource"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#personal-preference">
  <rdfs:subPropertyOf>
    <owl:ObjectProperty rdf:ID="personal-context-feature"/>
  </rdfs:subPropertyOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="is-in-context">
  <rdfs:range rdf:resource="#OrganizationalContextEntity"/>
  <rdfs:subPropertyOf rdf:resource="#organizational-context-feature"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="has-operating-system">
  <rdfs:subPropertyOf rdf:resource="#technical-context-feature"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="has-browser-plugin">
  <rdfs:subPropertyOf rdf:resource="#technical-context-feature"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="has-competency">
  <rdfs:subPropertyOf>
    <owl:ObjectProperty rdf:about="#personal-context-feature"/>
  </rdfs:subPropertyOf>
  <rdfs:range rdf:resource="#Competency"/>
  <rdfs:domain rdf:resource="#Person"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="is-subordinate-of">
  <rdfs:subPropertyOf rdf:resource="#social-context-feature"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="has-browser">
  <rdfs:subPropertyOf rdf:resource="#technical-context-feature"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="likes">
  <rdfs:subPropertyOf>
    <owl:ObjectProperty rdf:ID="knows"/>
  </rdfs:subPropertyOf>

```

```

</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#knows">
  <rdfs:subPropertyOf rdf:resource="#social-context-feature"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="prefers-semantic-density">
  <rdfs:subPropertyOf rdf:resource="#personal-preference"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="has-objective">
  <rdfs:range rdf:resource="#Competency"/>
  <rdfs:domain rdf:resource="#LearningObject"/>
  <rdfs:subPropertyOf rdf:resource="#learning-object-property"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#personal-context-feature">
  <rdfs:subPropertyOf rdf:resource="#context-feature"/>
</owl:ObjectProperty>
<owl:TransitiveProperty rdf:ID="subsumes">
  <rdfs:range rdf:resource="#Competency"/>
  <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
  <rdfs:domain rdf:resource="#Competency"/>
</owl:TransitiveProperty>
<owl:TransitiveProperty rdf:ID="is-higher-than">
  <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
  <rdfs:domain rdf:resource="#CompetencyLevel"/>
  <rdfs:range rdf:resource="#CompetencyLevel"/>
</owl:TransitiveProperty>
<owl:FunctionalProperty rdf:ID="at-level">
  <rdfs:domain rdf:resource="#Competency"/>
  <rdfs:range rdf:resource="#CompetencyLevel"/>
  <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
</owl:FunctionalProperty>
<swrl:Variable rdf:ID="x"/>
<DreyfusLevels rdf:ID="expert">
  <is-higher-than>
    <DreyfusLevels rdf:ID="competent">
      <is-higher-than>
        <DreyfusLevels rdf:ID="intermediate">
          <is-higher-than>
            <DreyfusLevels rdf:ID="beginner">
              <is-higher-than>
                <DreyfusLevels rdf:ID="novice"/>
              </is-higher-than>
            </DreyfusLevels>
          </is-higher-than>
        </DreyfusLevels>
      </is-higher-than>
    </DreyfusLevels>
  </is-higher-than>
</DreyfusLevels>
<swrl:Variable rdf:ID="y"/>
<swrl:Imp rdf:ID="CompetencySubsumption">
  <swrl:body>
    <swrl:AtomList>
      <rdf:first>
        <swrl:IndividualPropertyAtom>
          <swrl:argument1 rdf:resource="#x"/>
          <swrl:propertyPredicate rdf:resource="#has-competency"/>
          <swrl:argument2 rdf:resource="#y"/>
        </swrl:IndividualPropertyAtom>
      </rdf:first>
      <rdf:rest>
        <swrl:AtomList>
          <rdf:rest>
            <swrl:AtomList>
              <rdf:first>
                <swrl:IndividualPropertyAtom>
                  <swrl:argument1>
                    <swrl:Variable rdf:ID="z"/>
                  </swrl:argument1>
                  <swrl:argument2>
                    <swrl:Variable rdf:ID="w"/>
                  </swrl:argument2>
                  <swrl:propertyPredicate rdf:resource="#is-higher-than"/>
                </swrl:IndividualPropertyAtom>
              </rdf:first>
            </swrl:AtomList>
          </rdf:rest>
        </swrl:AtomList>
      </rdf:rest>
    </swrl:AtomList>
  </swrl:body>

```

```

    <rdf:rest rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#nil"/>
  </swrl:AtomList>
</rdf:rest>
<rdf:first>
  <swrl:IndividualPropertyAtom>
    <swrl:argument2 rdf:resource="#w"/>
    <swrl:argument1 rdf:resource="#y"/>
    <swrl:propertyPredicate rdf:resource="#at-level"/>
  </swrl:IndividualPropertyAtom>
</rdf:first>
</swrl:AtomList>
</rdf:rest>
</swrl:AtomList>
</swrl:body>
<swrl:head>
  <swrl:AtomList>
    <rdf:first>
      <swrl:IndividualPropertyAtom>
        <swrl:propertyPredicate rdf:resource="#has-competency"/>
        <swrl:argument2 rdf:resource="#z"/>
        <swrl:argument1 rdf:resource="#x"/>
      </swrl:IndividualPropertyAtom>
    </rdf:first>
    <rdf:rest rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#nil"/>
  </swrl:AtomList>
</swrl:head>
</swrl:Imp>
</rdf:RDF>

```

<!-- Created with Protege (with OWL Plugin 2.2 beta, Build 288) <http://protege.stanford.edu> -->

B.

Schema für die relationale Datenbank des Kontextfaktendienstes

```
CREATE TABLE users
(
    userid VARCHAR2(255) PRIMARY KEY
);

CREATE SEQUENCE factseq;

CREATE SEQUENCE featureseq;

CREATE TABLE datatypes
(
    typeid VARCHAR2(255) PRIMARY KEY
);

INSERT INTO datatypes(typeid) VALUES('string');
INSERT INTO datatypes(typeid) VALUES('number');
INSERT INTO datatypes(typeid) VALUES('date');
INSERT INTO datatypes(typeid) VALUES('ontology');

CREATE TABLE feature
(
    featureid INTEGER PRIMARY KEY,
    uri VARCHAR2(255) NOT NULL,
    "type" VARCHAR2(255) NOT NULL,
    agingfunction VARCHAR(255),
    agingparam1 FLOAT,
    agingparam2 FLOAT,
    agingparam3 FLOAT,
    cardinality INTEGER NOT NULL
);

ALTER TABLE feature ADD CONSTRAINT fk_feature_datatypes
FOREIGN KEY ("type") REFERENCES datatypes(typeid);

CREATE UNIQUE INDEX featureURIIndex ON feature(uri);

CREATE TABLE featurehierarchy
(
    parent INTEGER NOT NULL,
    child INTEGER NOT NULL
);

ALTER TABLE featurehierarchy ADD CONSTRAINT pk_hierarchy
PRIMARY KEY (parent,child);
ALTER TABLE featurehierarchy ADD CONSTRAINT fk_hierarchy1
FOREIGN KEY (parent) REFERENCES feature(featureid)
```

Anhang B. Schema für die relationale Datenbank des Kontextfaktendienstes

```
ON DELETE CASCADE;
ALTER TABLE featurehierarchy ADD CONSTRAINT fk_hierarchy2
FOREIGN KEY (child) REFERENCES feature(featureid)
ON DELETE CASCADE;

- Context Facts

CREATE TABLE numericalcontextfact
(
  id INTEGER PRIMARY KEY,
  featureid INTEGER NOT NULL,
  userid VARCHAR2(255) NOT NULL,
  validfrom TIMESTAMP,
  validuntil TIMESTAMP,
  confidence FLOAT NOT NULL,
  op VARCHAR2(10) NOT NULL,
  value NUMBER NOT NULL,
  transactiontime TIMESTAMP NOT NULL,
  transformedConfidence FLOAT NOT NULL
);

ALTER TABLE numericalcontextfact ADD CONSTRAINT fk_ncontextfact_feature
FOREIGN KEY (featureid) REFERENCES feature(featureid) ON DELETE CASCADE;
ALTER TABLE numericalcontextfact ADD CONSTRAINT fk_ncfact_user
FOREIGN KEY (userid) REFERENCES users(userid) ON DELETE CASCADE;

CREATE INDEX numconfidenceIndex ON numericalcontextfact(transformedConfidence);
CREATE INDEX numUserIndex ON numericalcontextfact(userid);
CREATE INDEX numFeatureIndex ON numericalcontextfact(featureid);
CREATE INDEX numValidIndex ON numericalcontextfact(validFrom,validUntil);

CREATE TABLE ontologycontextfact
(
  id INTEGER PRIMARY KEY,
  featureid INTEGER NOT NULL,
  userid VARCHAR2(255) NOT NULL,
  validfrom TIMESTAMP,
  validuntil TIMESTAMP,
  confidence FLOAT NOT NULL,
  op VARCHAR2(10) NOT NULL,
  value VARCHAR2(255) NOT NULL,
  ontology VARCHAR2(255) NOT NULL,
  source VARCHAR2(255) NOT NULL,
  transactiontime TIMESTAMP NOT NULL,
  transformedConfidence FLOAT NOT NULL
);

ALTER TABLE ontologycontextfact ADD CONSTRAINT fk_ontologycontextfact_feature
FOREIGN KEY (featureid) REFERENCES feature (featureid) ON DELETE CASCADE;
ALTER TABLE ontologycontextfact ADD CONSTRAINT fk_ocfact_user
FOREIGN KEY (userid) REFERENCES users(userid) ON DELETE CASCADE;

CREATE INDEX ontconfidenceIndex ON ontologycontextfact(transformedConfidence);
CREATE INDEX ontUserIndex ON ontologycontextfact(userid);
CREATE INDEX ontFeatureIndex ON ontologycontextfact(featureid);
CREATE INDEX ontValidIndex ON ontologycontextfact(validFrom,validUntil);

CREATE TABLE stringcontextfact
(
  id INTEGER PRIMARY KEY,
  featureid INTEGER NOT NULL,
  userid VARCHAR2(255) NOT NULL,
  validfrom TIMESTAMP,
  validuntil TIMESTAMP,
  confidence FLOAT NOT NULL,
  op VARCHAR2(10) NOT NULL,
  value VARCHAR2(255) NOT NULL,
  source VARCHAR2(255) NOT NULL,
  transactiontime TIMESTAMP NOT NULL,
  transformedConfidence FLOAT NOT NULL
);

ALTER TABLE stringcontextfact ADD CONSTRAINT fk_stringcontextfact_feature
FOREIGN KEY (featureid) REFERENCES feature (featureid) ON DELETE CASCADE;
```

```

ALTER TABLE stringcontextfact ADD CONSTRAINT fk_scfact_user
  FOREIGN KEY (userid) REFERENCES users(userid) ON DELETE CASCADE;

CREATE INDEX stringconfidenceIndex ON stringcontextfact(transformedConfidence);
CREATE INDEX stringconfidenceUserIndex ON stringcontextfact(userid);
CREATE INDEX stringFeatureIndex ON stringcontextfact(featureid);
CREATE INDEX stringValidIndex ON stringcontextfact(validFrom,validUntil);

CREATE TABLE datecontextfact
(
  id INTEGER PRIMARY KEY,
  featureid INTEGER NOT NULL,
  userid VARCHAR2(255) NOT NULL,
  validfrom TIMESTAMP,
  validuntil TIMESTAMP,
  confidence FLOAT NOT NULL,
  op VARCHAR2(10) NOT NULL,
  value TIMESTAMP NOT NULL,
  source VARCHAR2(255),
  transactiontime TIMESTAMP NOT NULL,
  transformedConfidence FLOAT NOT NULL
);

ALTER TABLE datecontextfact ADD CONSTRAINT fk_datecontextfact_feature
  FOREIGN KEY (featureid) REFERENCES feature(featureid) ON DELETE CASCADE;
ALTER TABLE datecontextfact ADD CONSTRAINT fk_dcfact_user
  FOREIGN KEY (userid) REFERENCES users(userid) ON DELETE CASCADE;

CREATE INDEX dateconfidenceIndex ON datecontextfact(transformedConfidence);
CREATE INDEX dateconfidenceUserIndex ON datecontextfact(userid);
CREATE INDEX dateFeatureIndex ON datecontextfact(featureid);
CREATE INDEX dateValidIndex ON datecontextfact(validFrom,validUntil);

create or replace trigger feature_auto
  before insert on feature
  for each row
  begin
    select factseq.nextval into :new.featureid from dual;
  end;
/

create or replace trigger ncontextfact_auto
  before insert on numericalcontextfact
  for each row
  begin
    select factseq.nextval into :new.id from dual;
  end;
/

create or replace trigger ocontextfact_auto
  before insert on ontologycontextfact
  for each row
  begin
    select factseq.nextval into :new.id from dual;
  end;
/

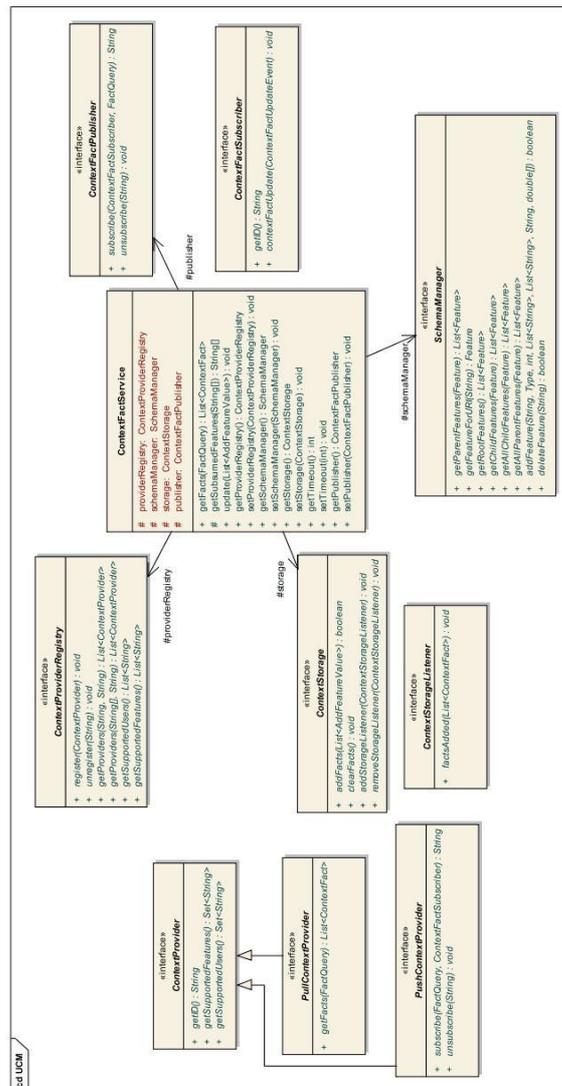
create or replace trigger stringcontextfact_auto
  before insert on stringcontextfact
  for each row
  begin
    select factseq.nextval into :new.id from dual;
  end;
/

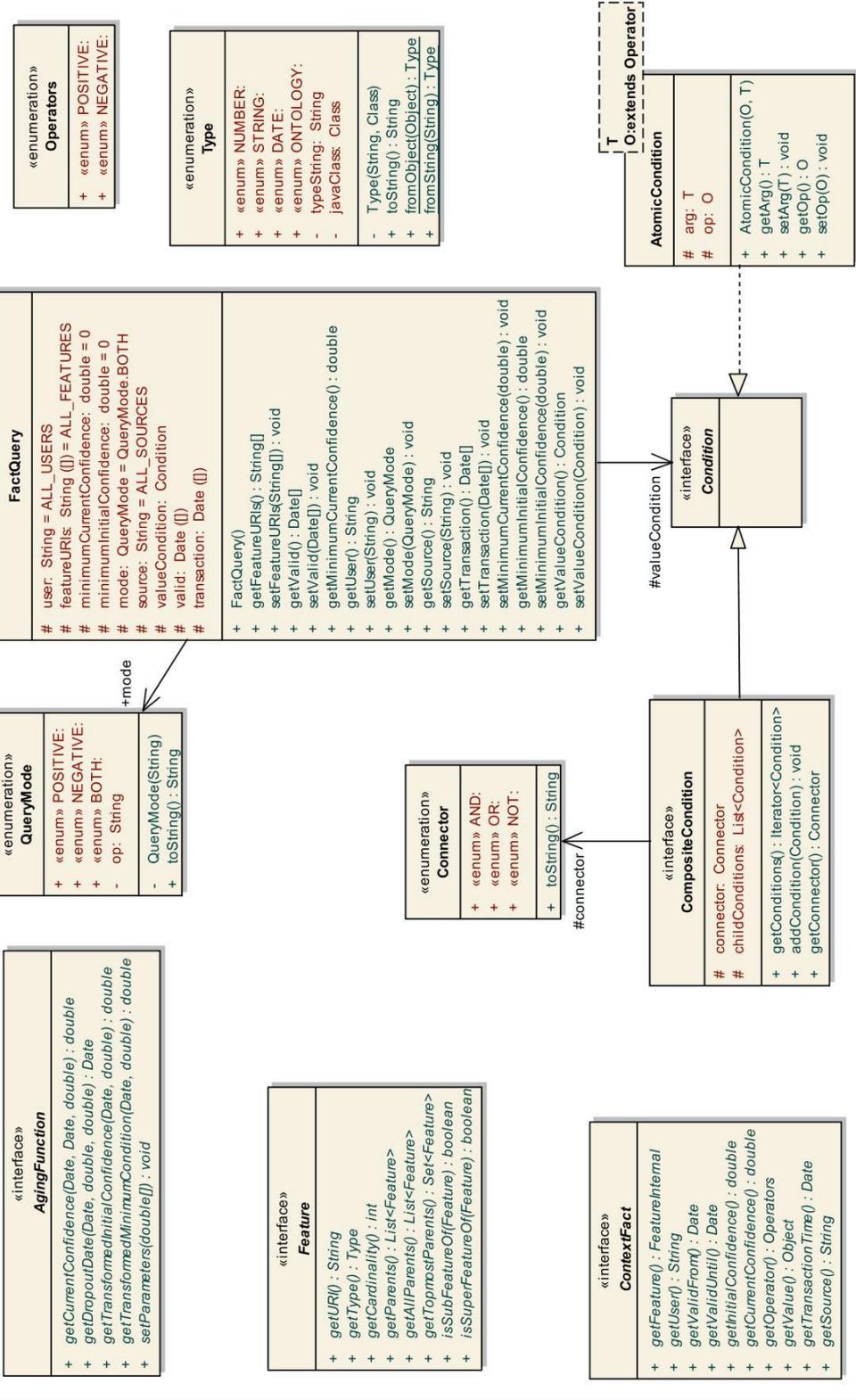
create or replace trigger datecontextfact_auto
  before insert on datecontextfact
  for each row
  begin
    select factseq.nextval into :new.id from dual;
  end;
/

```


C.

UML-Diagramme Kontextverwaltungsdienst







Beispiel-Protokoll formative Evaluierung

9
10
11 LIP system scenario
12
13 Name: p9
14
15 Job title: Consultant
16
17 Company:
18
19 Context
20 I am [name] I'm integrated in the Java Centre and I am involved in many projects dealing with
21 Tourism as analyst and developer.
22
23
24
25 Goal
26 I am developing a Java application and I have a problem that is the following: I need to
27 communicate the current application that I am developing with a database. And I do not know
28 how to do it.
29
30
31 User role and task(s):
32
33 Learner
34 Searching for elearning objects or programmes
35 Consume elearning objects
36 Communicating with an expert
37 Joining a learning group
38 Consume elearning objects suggested by system to help current work task
39
40
41 Action
42 First day
43 1- I open the application and there is a screen to login. I introduce my password and
44 username, and then appear my interface with the FrontOffice.
45 2- In the interface I have an input test [text] box where to search eLo objects or programmes
46 about specific subject.
47 3- I introduce Java and Database
48 4- After some seconds, it appears on the screen a list of eLearning objects o eLearning
49 Programs with a small description of their content.
50 5- I select one of the objects that is about JDBC -Java Database Connectivity
51 6- It appears a description on what is the e-learning object and what it servers for, the time
52 consumption, etc. , apart from the description of the object I would like to have some kind
53 of comparative table about other possible eLobject's - projects related with this specific one
54 7- I select the programme and the system asks me about my preferences on this specific
55 object/programme as for example, I want to be asked about the level of deepness and the
56 types of files that I want to see (In this case, I want to just be introduced to the subject
57 and I prefer to just see multimedia or power point files).
58 8- The system answer that it is not possible to accomplish all my preferences, it is possible to
59 just be introduced to the subject but it is in a word document. And it asks me if I want to

Anhang D. Beispiel-Protokoll formative Evaluierung

60 be shown with the content.
61 9- I select yes, and I consume the content presented. And I logout of the system.
62
63 Some time has passed.
64
65 Second day
66 1. I open the application, I introduce the username and password.
67 2. I do another search on Java and DDBB.
68 3. Apart from the list of e-Learning Objects and e-learning programmes on the subject. It
69 appears in the first place of the list and highlighted a e-LObject that is the natural
70 continuation of the e-Learning object. (In this case, a manual about how to develop the
71 functionality to connect a DDBB with a Java application).
72 4. I select this e-Learning Programme, and it is compiled according to my context and
73 preferences.
74 5. Once consumed I would like to be offered with the possibility of taking or not taking a test
75 on the subject.
76 6. I logout the system.
77
78 Third day
79
80 1. I open the application, I introduce the username and password.
81 2. I start working with my daily applications.
82 3. Then the system detects that I am having some difficulties with the application
83 4. It offers me to take a e-LObject with a small description that could help me in dealing with
84 the application and it offers me a list of users that are expert on the subject matter.
85 5. I decide to check the extended description of the object (time consumption, what it is for,
86 etc.). After reading it I decide that I prefer to communicate with one of the experts.
87 6. I come back to the previous screen and I select on [one] of the experts. Than it appears a
88 small window (as a messenger system) where I introduce my doubt.
89 7. If the expert is connected, it appears on his/her desktop a window with my doubt and he is
90 able to write down the reply. If he is not connected, the small window with my doubt (and
91 my identification) will appear next time he logs into the LIP system. He will reply and I will
92 be notified by the LIP system when the reply has been received.
93 8. The system suggests me to join a learning group about the subject I am working on DDBB
94 connectivity with Java applications.
95 9. I decide to join the group, then a window appear to register to the group. I just accept and
96 the rest of members are notified (in the window type messenger) that a new member has
97 joined.
98 10. I Logout the system
99
100 Fourth day
101
102 1. I open the application, I introduce the username and password.
103 2. I start working with my daily applications.
104 3. The system automatically suggests me a e-Learning Object related to my current tasks.
105 4. I select and consume it.
106 5. When having finished. I want to rate the e-learning Object in and give some text feedback,
107 about the context and the structure. I do it in a screen where there is a 1-10 score and
108 above there is a text box where I write. I do so and after I logout.
109
110

Literaturverzeichnis

- [ABD⁺01] ABECKER, Andreas ; BERNARDI, Ansgar ; DIOUDIS, Spyros ; ELST, Ludger van ; HERTERICH, Rudi ; HOUY, Christian ; LEGAL, Maria ; MENTZAS, Gregory ; MÜLLER, Stephan: Workflow-Embedded Organizational Memory Access: The DECOR Project. In: *International Conference on Artificial Intelligence (IJCAI), Workshop on Knowledge Management and Organizational Memories, Seattle, 2001*
- [Abe04] ABECKER, Andreas: *Business process oriented knowledge management: concepts, methods, and tools*, Fakultät für Wirtschaftswissenschaften, Universität Karlsruhe, Diss., 2004
- [ABS99] ABECKER, Andreas ; BERNARDI, Ansgar ; SINTEK, M.: Proactive Knowledge Delivery for Enterprise Knowledge Management. In: RUHE, G. (Hrsg.) ; BOMARIUS, F. (Hrsg.): *Learning Software Organizations, Methodology and Applications, 11th International Conference on Software Engineering and Knowledge Engineering, SEKE'99, Kaiserslautern, Germany, June 16-19, 1999*, Springer, 1999, S. 103–117
- [Adv04] ADVANCED DISTRIBUTED LEARNING (ADL) INITIATIVE: Sharable Content Object Reference Model (SCORM) Content Aggregation Model (CAM) Version 1.3 / ADL. 2004. – Forschungsbericht
- [Aes04] AESCHLIMANN, Thomas: *Skill-Management im Geschäftsbereich Verkehr der Schweizerischen Bundesbahnen*, Wirtschafts- und Sozialwissenschaftliche Fakultät, Universität Bern, Lizentiatsarbeit, 2004
- [AF00] ARTALE, A. ; FRANCONI, E.: A Survey of Temporal Extensions of Description Logics. In: *Annals of Mathematics and Artificial Intelligence* 30 (2000), S. 171–210
- [AFGD02] ALMEIDA FALBO, Ricardo de ; GUIZZARDI, Giancarlo ; DUARTE, Katia C.: An Ontological Approach to Domain Engineering. In: *International Conference on Software Engineering and Knowledge Engineering (SEKE 02)*, 2002

- [AFWZ02] ARTALE, A. ; FRANCONI, E. ; WOLTER, F. ; ZAKHARYASHEV, M.: A Temporal Description Logic for Reasoning over Conceptual Schemas and Queries. In: *8th European Conference on Logics in Artificial Intelligence (JELIA-02)*. Cosenza, Italy : Springer, 2002
- [AHMM02] ABECKER, Andreas (Hrsg.) ; HINKELMANN, Knut (Hrsg.) ; MAUS, Heiko (Hrsg.) ; MÜLLER, Hans J. (Hrsg.): *Geschäftsprozessorientiertes Wissensmanagement - Effektive Wissensnutzung bei der Planung und Umsetzung von Geschäftsprozessen*. Springer, 2002
- [AL06] ATEYEH, Khaldoun ; LOCKEMANN, Peter C.: Reuse- and Aspect-Oriented Courseware Development. In: *Educational Technology & Society* 9 (2006), Nr. 4, S. 95–113
- [All01] ALLEN, Chuck: Competencies 1.0 (Measurable Characteristics) / HR-XML. 2001. – Forschungsbericht
- [All04] ALLEN, Chuck: HR-XML Competencies (Measurable Characteristics) / HR-XML Consortium. 2004. – Forschungsbericht
- [Ams78] AMSTAD, Tony: *Wie verständlich sind unsere Zeitungen?*, Universität Zürich, Diss., 1978
- [AN95] AAMODT, A. ; NYGARD, M.: Different roles and mutual dependencies of data, information, and knowledge - an AI perspective on their integration. In: *Data and Knowledge Engineering* 16 (1995), S. 191–222
- [Ang03] ANGHERN, Albert A.: ICDT Platform: An Overview / INSEAD CALT. Version: 2003. http://www.calt.insead.edu/project/ICDT-Platform/documents/2003-ICDT-Platform_overview.pdf. 2003. – Forschungsbericht
- [APD91] ARANGO, G. ; PRIETO-DIAZ, R.: Domain Analysis Concepts and Research Directions. In: *Domain Analysis and Software Systems Engineering*. IEEE Computer Society Press, 1991
- [AS78] ARGYRIS, C. ; SCHÖN, D.: *Organizational Learning: A Theory of Action Perspective*. Addison Wesley, 1978
- [Att07] ATTWELL, Graham: The Personal Learning Environments - the future of eLearning? In: *eLearning Papers* 2 (2007), Nr. 1. – ISSN 1887–1542

- [Bar02] BARRETT, Helen: *Pedagogical Issues in Electronic Portfolio Systems*. Online. <http://electronicportfolios.com/EPpedissues.pdf>. Version: Oct 2002
- [Bat86] BATES, Marcia J.: An exploratory paradigm for online information retrieval. In: BROOKES, B.C. (Hrsg.): *Intelligent Information Systems for the Information Society*. Amsterdam : North-Holland, 1986
- [Bat89] BATES, M.J.: The design of browsing and berrypicking techniques for the on-line search interface. In: *Online Review* 13 (1989), Nr. 5, S. 407–431
- [Bat02] BATES, Marcia J.: Toward an Integrated Model of Information Seeking and Searching. In: *Fourth International Conference on Information Needs, Seeking and Use in Different Contexts, Lisbon, Portugal, September 11-13, 2002*, 2002
- [Bat05] BATES, Marcia J.: Information and knowledge: an evolutionary framework for information science. In: *Information Research* 10 (2005), Nr. 4, S. paper 239
- [BB05] BAZIRE, Mary ; BRÉZILLON, Patrick: Understanding Context Before Using It. In: *Modeling and Using Context: 5th International and Interdisciplinary Conference CONTEXT 2005, Paris, France, July 5-8, 2005* Bd. 3554, Springer, 2005 (Lecture Notes in Computer Science)
- [BBSS01] BACK, Andrea ; BENDEL, Oliver ; STOLLER-SCHAI, Daniel: *E-Learning im Unternehmen. Grundlagen - Strategien - Methoden - Technologien*. Zürich : Orell Füssli, 2001
- [BC05] BARRETT, Helen ; CARNEY, Joanne: Conflicting Paradigms and Competing Purposes in Electronic Portfolio Development. In: *Educational Assessment* (2005)
- [BD00] BROWN, John S. ; DUGUID, Paul: *The Social Life of Information*. Harvard Business School Press, 2000
- [BD01] BIMBÓ, Katalin ; DUNN, J. M.: Four-valued Logic. In: *Notre Dame Journal on Formal Logic* 42 (2001), Nr. 3, S. 171–192
- [BDE⁺04] BERNARDI, Ansgar ; DENGEL, Andreas ; ELST, Ludger van ; LAUER, Andreas ; MAUS, Heiko ; SCHWARZ, Sven: FRODO: Ein Agentenframework für verteilte Unternehmensgedächtnisse. In: *KI* 18 (2004), Nr. 3, S. 59–
- [Bel77] BELNAP, Nuel D.: A useful four-valued logic. In: *Modern Uses of Multiple-Valued Logic*. Dordrecht : Reidel, 1977

- [Bel80] BELKIN, N.: Anomalous States of Knowledge as a basis for information retrieval. In: *Canadian Journal of Information Science* 1980 (1980), Nr. 5, S. 133–143
- [Ben02] BENDEL, Oliver: Pädagogische Agenten im Corporate E-Learning. In: NEUMANN, Reiner (Hrsg.) ; NACKE, Ralf (Hrsg.) ; ROSS, Alexander (Hrsg.): *Corporate E-Learning: Strategien, Märkte, Anwendungen*. Wiesbaden : Gabler, 2002, S. 97–106
- [Ben03] BENDEL, Oliver: *Pädagogische Agenten im Corporate E-Learning*. Bamberg : Difo-Druck, 2003
- [Ben04] BENDEL, Oliver: Merkmale, Ziele und Funktionen pädagogischer Agenten. In: BEKAVAC, Bernard (Hrsg.) ; HERGEL, Josef (Hrsg.) ; RITTBERGER, Marc (Hrsg.): *Informationen zwischen Kultur und Marktwirtschafts. Proceedings des 9. Internationalen Symposiums für Informationswissenschaften (ISI 2004)*, Chur. Konstanz : UVK Verlagsgesellschaft, 2004, S. 213–226
- [Ber02] BERZTISS, Alfs T.: Uncertainty Management. In: CHANG, Shi-Kuo (Hrsg.): *Handbook of Software Engineering and Knowledge Engineering* Bd. 2. World Scientific, 2002
- [BGMP92] BARBARÁ, D. ; GARCÍA-MOLINA, H. ; PORTER, D.: The Management of Probabilistic Data. In: *ACM Transactions on Knowledge and Data Engineering* 4 (1992), Oct, Nr. 5, S. 487–502
- [BGMU06] BROOKS, C. ; GREER, J. ; MELIS, E. ; ULLRICH, C.: Combining ITS and eLearning Technologies: Opportunities and Challenges. In: *th International Conference on Intelligent Tutoring Systems (ITS2006)*, June 26 – June 30, 2006. Jhongli, Taiwan, 2006
- [BH04] BENDEL, Oliver ; HAUSKE, Stefanie: *E-Learning: Das Wörterbuch*. Sauerländer Verlage, 2004
- [BH05] BERIO, Guiseppa ; HARZALLAH, Mounira: Knowledge Management for Competence Management. In: *Journal of Universal Knowledge Management* 0 (2005), Nr. 1, S. 21–38
- [BHS07] BRAUN, Simone ; HEFKE, Mark ; SCHMIDT, Andreas: A Socially-Aware Desktop for e-Science: Supporting Learning in Networked Scientific Processes. In: GRONAU, Norbert (Hrsg.): *4th Conference Professional Knowledge Management - Experiences and Visions (WM '07)*, Potsdam Bd. 2. Berlin : GITO-Verlag, 2007. – ISBN 978-3-936771-99-2, 47-54

- [Bie06] BIESALSKI, Ernst: *Unterstützung der Personalentwicklung mit ontologie-basiertem Kompetenzmanagement*, Fakultät für Wirtschaftswissenschaften, Universität Karlsruhe, Diss., 2006
- [BKPS04] BÖCKLE, G. (Hrsg.) ; KNAUBER, P. (Hrsg.) ; POHL, K. (Hrsg.) ; SCHMID, K. (Hrsg.): *Software-Produktlinien*. dpunkt.Verlag, 2004
- [BKS03] BUCHHOLZ, Thomas ; KÜPPER, Axel ; SCHIFFERS, Michael: *Quality of Context: What It Is And Why We Need It*. In: *10th International Workshop of the HP OpenView University Association (HPOVUA 2003), Geneva, Switzerland, 2003*
- [BKSS08] BALFANZ, Dirk ; KLEIN, Michael ; SCHMIDT, Andreas ; SANTI, Marco: *Partizipative Entwicklung einer Middleware für AAL-Lösungen*. In: *GMS Medizinische Informatik, Biometrie und Epidemiologie 4* (2008), Nr. 3. <http://www.egms.de/en/journals/mibe/2008-4/mibe000078.shtml>
- [BL93] BATTISTA, Giuseppe di ; LENZERINI, Maurizio: *Deductive entity relationship modeling*. In: *IEEE Transactions on Knowledge and Data Engineering 5* (1993), Nr. 3, S. 439–450
- [BLRT03] BUNSCHKOWSKI, M. ; LUCKE, U. ; RÖSER, M. ; TAVANGARIAN, D.: *The role of metadata and the automated generation of educational material*. In: *2nd European Conference on e-Learning, England, 2003*, S. 93–104
- [Blu69] BLUMER, Herbert: *Der methodologische Standort des Symbolischen Interaktionismus*. In: *Alltagswissen, Interaktion und gesellschaftliche Wirklichkeit. Band I*. Westdeutscher Verlag, 1969, S. 80–141
- [Blu73] BLUMER, Herbert: *Der methodologische Standort des symbolischen Interaktionismus*. In: SOZIOLOGEN, Arbeitsgruppe B. (Hrsg.): *Alltagswissen, Interaktion und gesellschaftliche Wirklichkeit*. Rowohlt, 1973, S. 80–146
- [Blu98] BLUMSTENGEL, Astrid: *Entwicklung hypermedialer Lernsysteme*. Wissenschaftlicher Verlag, Berlin, Universität Paderborn, Dissertation, 1998. <http://dsor.unipaderborn.de/de/forschung/publikationen/blumstengel-diss/>
- [BN06] BIELIKOVA, Maria ; NAGY, Peter: *Considering Human Memory Aspects for Adaptation and Its Implementation in AHA!* In: *European Conference on Technology-Enhanced Learning (ECTEL 06), Crete, 2006*

- [BO01] BACK, Jonathan ; OPPENHEIM, Charles: A model of cognitive load for IR: implications for user relevance feedback interaction. In: *Information Research* 6 (2001), Nr. 2
- [BP93] BOSC, P. ; PRADE, H.: An Introduction to Fuzzy Set and Possibility Theory Based Approaches to the Treatment of Uncertainty and Imprecision in Database Management Systems. In: *2nd Workshop on Uncertainty Management in Information Systems, Catalina*, 1993
- [BP03] BARETT, Kearra ; POWER, Ruadhri: State of the Art: Context Management / m-Zones. 2003 (1). – m-Zones deliverable
- [Bré99] BRÉZILLON, Patrick: Context in problem solving: A survey. In: *The Knowledge Engineering Review* 14 (1999), Nr. 1, S. 1–34
- [BR03] BLACKMON, William ; REHAK, Daniel: Customized Learning: A Web Services Approach. In: *World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA 03)*, 2003
- [Bro96] BROWN, P.J.: The Stick-e Document: A Framework For Creating Context-aware Applications. In: *Proceedings of the Electronic Publishing*. Laxenburg, Austria, Sep 1996, S. 259–272
- [Bru98] BRUSILOVSKY, Peter: Adaptive Educational Systems on the World-Wide-Web: A Review of Available Technologies. In: *4th Conference on Intelligent Tutoring Systems*, 1998
- [BS85] BEREITER, C. ; SCARDAMALIA, M.: Cognitive Coping Strategies and the Problem of 'Inert Knowledge'. In: CHIPMAN, S.F. (Hrsg.) ; SEAGAL, J.W. (Hrsg.) ; GLASER, R. (Hrsg.): *Thinking and Learning Skills* Bd. 2. Hillside, NJ, USA : LEA, 1985
- [BS99] BERGMANN, Bärbel ; SONNTAG, Karlheinz: Die Umsetzung und Generalisierung erworbener Kompetenzen in den Arbeitsalltag. In: SONNTAG, Karlheinz (Hrsg.): *Personalentwicklung in Organisationen*. Göttingen : Hogrefe, 1999, S. 287–312
- [BS02a] BETTONI, Marco C. ; SCHNEIDER, Sibylle: The Essence of Knowledge Management: A constructivist approach. In: *Knowledge Management 2002, London, U.K.*, 2002
- [BS02b] BIRKINSHAW, Julian ; SHEEHAN, Tony: Managing the Knowledge Life Cycle. In: *Sloan Management Review* 44 (2002), Nr. 1, S. 75–83

- [BS06a] BRAUN, Simone ; SCHMIDT, Andreas: Don't Annoy the Informal Teacher: Context-Aware Mediation of Communication for Workplace Learning. In: *6th International Conference on Knowledge Management (IKNOW '06), Special Track on Integrating Working and Learning in Business*, 2006
- [BS06b] BRAUN, Simone ; SCHMIDT, Andreas: Socially-Aware Informal Learning Support: Potentials and Challenges of the Social Dimension. In: *2nd International Workshop on Learner-Oriented Knowledge Management and KM-Oriented Learning (LOKMOL 06) in conjunction with the 1st European Conference on Technology-Enhanced Learning (ECTEL06), Heraklion, Crete*, 2006
- [BS07] BRAUN, Simone ; SCHMIDT, Andreas: Wikis as a Technology Fostering Knowledge Maturing: What we can learn from Wikipedia. In: *7th International Conference on Knowledge Management (IKNOW '07), Special Track on Integrating Working and Learning in Business (IWL)*, 2007
- [BS08] BRAUN, Simone ; SCHMIDT, Andreas: People Tagging & Ontology Maturing: Towards Collaborative Competence Management. In: *8th International Conference on the Design of Cooperative Systems (COOP '08), Carry-le-Rouet, France, May 20-23, 2008*, 2008
- [BSS01] BECKER, Matthias ; SPÖTTL, Georg ; STOLTE, Andre: Neue Lernmodelle - Flexible und akzeptierte Wege zum Lernen für die Arbeitswelt / Nationale Unterstüztungsstelle ADAPT der Bundesanstalt für Arbeit. 2001. – Forschungsbericht
- [BSW⁺07] BRAUN, Simone ; SCHMIDT, Andreas ; WALTER, Andreas ; NAGYPAL, Gabor ; ZACHARIAS, Valentin: Ontology Maturing: a Collaborative Web 2.0 Approach to Ontology Engineering. In: *Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge at the 16th International World Wide Web Conference (WWW 07), Banff, Canada*, 2007
- [BSWZ07] BRAUN, Simone ; SCHMIDT, Andreas ; WALTER, Andreas ; ZACHARIAS, Valentin: The Ontology Maturing Approach to Collaborative and Work-Integrated Ontology Development: Evaluation Results and Future Directions. In: *International Workshop on Emergent Semantics and Ontology Evolution (ESOE), ISWC 2007, Busan/Korea*, 2007
- [BT85] BONNISSONE, P. ; TONG, R.: Reasoning with Uncertainty in Expert Systems. In: *International Journal of Man Machine Studies* 22 (1985), S. 241–250

- [BV03] BRUSILOVSKY, Peter ; VASSILEVA, J.: Course Sequencing for Large-Scale Web-based Education. In: *International Journal of Continuing Engineering Education and Life-long Learning* 13 (2003), Nr. 1/2, S. 75–94
- [BW01] BABU, Shivnath ; WIDOM, Jennifer: Continuous queries over data streams. In: *SIGMOD Rec.* 30 (2001), Nr. 3, S. 109–120. <http://dx.doi.org/http://doi.acm.org/10.1145/603867.603884>. – DOI <http://doi.acm.org/10.1145/603867.603884>. – ISSN 0163–5808
- [Car95] CARROLL, J.: *Scenario-Based Design: Envisioning work and Technology in System Development* Wiley1995. Wiley, 1995
- [CB03] COOK, J. ; BRADLEY, C.: Creating More Effective User-Center E-Learning Systems Through Scenario-Based, Formative Evaluation. In: *2nd International Conference on Multimedia and Information and Communication Technologies in Education (m-ICTE 2003), Badajoz, Spain, 2003*
- [CBF04a] COOK, J. ; BRADLEY, C. ; FRANZOLINI, P.: Designing Effective eLearning Using the Scenario Based Evaluation Technique. In: *World Conference on Educational Multimedia, Hypermedia and Telecommunications (ED-MEDIA 04), Lugano, Switzerland, 2004*
- [CBF04b] COOK, John ; BRADLEY, Claire ; FRANZOLINI, Pablo: External Learner Evaluation Report / Learning in Process. 2004 (D8.3). – LIP Project Deliverable
- [CDES05] CHONG, Eugene I. ; DAS, Souripriya ; EADON, George ; SRINIVASAN, Jagannathan: An Efficient SQL-based RDF Querying Scheme. In: *Proceedings of the 31st VLDB Conference, Trondheim, Norway, 2005*, S. 1216–1227
- [CDT00] CHOO, Chun W. ; DETLOR, Brian ; TURNBULL, Don: Information Seeking on the Web: An Integrated Model of Browsing and Searching. In: *First Monday* 5 (2000), Feb, Nr. 2. http://firstmonday.org/issues/issue5_2/choo/index.html
- [CFJ04a] CHEN, Harry ; FININ, Tim ; JOSHI, Anupam: An Ontology for Context-Aware Pervasive Computing Environments. In: *TODO* Bd. 18, Cambridge University Press, May 2004, S. 197–207
- [CFJ04b] CHEN, Harry ; FININ, Timothy W. ; JOSHI, Anupam: Semantic Web in the Context Broker Architecture. In: *PerCom[DBL04]*, S. 277–286

- [Cha02] CHALMERS, Dan: *Contextual Mediation to Support Ubiquitous Computing*, Imperial College London, Diss., 2002
- [Che04] CHEN, Harry: *An Intelligent Broker Architecture for Pervasive Context-Aware Systems*, University of Maryland, PhD Thesis, 2004
- [Cho01] CHOO, Chun W.: Environmental scanning as information seeking and organizational learning. In: *Information Research* 7 (2001), Oct, Nr. 7
- [Cho02] *Kapitel Closing the Cognitive Gaps: How People Process Information*. In: CHOO, Chun W.: *Mastering Information Management*. FT-Prentice Hall, 2002
- [CK00] CHEN, G. ; KOTZ, D.: A Survey of Context-Aware Mobile Computing Research / Dartmouth Collegem Computer Science. 2000. – Forschungsbericht
- [CLMM07] CHRISTIAENS, Stijn ; LEENHEER, Pieter D. ; MOOR, Aldo D. ; MEERSMAN, Robert: Ontologising Competencies in an Interorganisational Setting. In: HEPP, Martin (Hrsg.) ; LEENHEER, Pieter D. (Hrsg.) ; MOOR, Aldo D. (Hrsg.) ; SURE, York (Hrsg.): *Ontology Management: Semantic Web, Semantic Web Services, and Business Applications*. Springer, 2007
- [CMHE04] COFFIELD, F. ; MOSELEY, D. ; HALL, E. ; ECCLESTONE, K.: Learning styles and pedagogy in post-16 learning. A systematic and critical review. / Learning and Skills Research Centre. Version: 2004. <http://www.lsd.org.uk/files/PDF/1543.pdf>. London, 2004. – Forschungsbericht
- [Con87] CONKLIN, Jeff: Hypertext: An Introduction and Survey. In: *IEEE Computer* 20 (1987), Nr. 9, S. 17–41
- [Coo00] COOPER, Kenneth C.: *Effective Competency Modeling and Reporting*. American Management Association, 2000
- [Cou01] COUNCIL OF EUROPE: *The Common European Framework of Reference for Languages*. Cambridge University Press, 2001
- [CP87] CAVALLO, Roger ; PITTARELLI, Michael: The Theory of Probabilistic Databases. In: *VLDB '87: Proceedings of the 13th International Conference on Very Large Data Bases*. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1987. – ISBN 0–934613–46–X, S. 71–81

- [CR02] COUTAZ, Joëlle ; REY, Gaëtan: Foundations for a Theory of Contextors. In: *CADUI 2002, May 15-17, Valenciennes, France, 2002*
- [CS91] CHANDLER, Paul ; SWELLER, John: Cognitive Load Theory and the Format of Instruction. In: *Cognition and Instruction* 8 (1991), S. 293 – 332. – ISSN 0737–0008
- [DBL04] *Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications (PerCom 2004), 14-17 March 2004, Orlando, FL, USA.* IEEE Computer Society, 2004 . – ISBN 0–7695–2090–1
- [DCES04] DAS, Souripriya ; CHONG, Eugene I. ; EADON, George ; SRINIVASAN, Jaggannathan: Supporting Ontology-based Semantic Matching in RDBMS. In: *Proceedings of the 30th VLDB Conference, Toronto, Canada, 2004*
- [DD86] DREYFUS, Hubert L. ; DREYFUS, S.: *Mind over Machine. The Power of Human Intuition and Expertise in the Era of the Computer.* New York : Basil Blackwell, 1986
- [Deh02] DEHNBOSTEL, Peter: Informelles Lernen – Aktualität und begrifflich-inhaltliche Einordnungen. In: *Informelles Lernen - eine Herausforderung für die berufliche Aus- und Weiterbildung.* Bielefeld : Bertelsmann, 2002, S. 3–12
- [Der99] DERVIN, Brenda: On studying information seeking methodologically: the implications of connecting metatheory to method. In: *Information Processing and Management* 35 (1999), Nr. 6, S. 727–750
- [Dey98] DEY, Anind K.: Context-aware computing: The CyberDesk project. In: *AAAI 1998 Spring Symposium on Intelligent Environments.* Palo Alto : AAAI Press., 1998, 51-54
- [Dey01] DEY, Anind K.: Understanding and Using Context. In: *Personal and Ubiquitous Computing Journal* 1 (2001), Nr. 5, S. 4–7
- [DGHH02] *Kapitel 11.* In: [AHMM02], S. 275–292
- [DHNS04] DOLOG, Peter ; HENZE, Nicola ; NEJDL, Wolfgang ; SINTEK, Michael: Personalization in Distributed eLearning Environments. In: *Proceedings of WWW 2004, 2004*
- [DJ92] DUFFY, T.M. ; JONASSEN, D.H.: Constructivism: New Implications for Instructional Technology. In: DUFFY, T.M. (Hrsg.) ; JONASSEN, D.H. (Hrsg.): *Constructivism and the Technology of Instruction: A Conversation.* Hillsdale NJ : Lawrence Erlbaum, 1992, S. 1–16

- [DM04] DENMAN-MAIER, Edith: Intercultural Factors in Web-based Training Systems. In: *Journal of Universal Computer Science* 10 (2004), Nr. 1, S. 80–104
- [DN03] DOLOG, P. ; NEJDL, W.: Challenges and Benefits of the Semantic Web for User Modelling. In: *AH2003 Workshop at WWW2003*, 2003
- [Doh01] DOHMEN, Günther: *Das informelle Lernen: Die internationale Erschließung einer bisher vernachlässigten Grundform menschlichen Lernens für das lebenslange Lernen aller*. Bundesministerium für Bildung und Forschung (BMBF), 2001
- [DRM98] DÖRING, K. ; RITTER-MAMCZEK, B.: *Die Praxis der Weiterbildung*. Weinheim, 1998
- [DRS01] DEKHTYAR, A. ; ROSS, R. ; SUBRAHMANIAN, V.S.: Probabilistic Temporal Databases I: Algebra. In: *ACM Transactions on Database Systems* 26 (2001), Nr. 1, S. 41–95
- [DS96] DEY, Debabrata ; SARKAR, Sumit: A probabilistic relational model and algebra. In: *ACM Transactions on Database Systems* 21 (1996), Nr. 3, S. 339–369. <http://dx.doi.org/http://doi.acm.org/10.1145/232753.232796>. – DOI <http://doi.acm.org/10.1145/232753.232796>. – ISSN 0362–5915
- [DS98] DYRESON, C. ; SNODGRASS, R.T.: Supporting Valid-Time Indeterminacy. In: *ACM Transactions on Database Systems* 23 (1998), Nr. 1, S. 1–57
- [DSA01] DEY, K. ; SALBER, D. ; ABOWD, G.D.: A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. In: *Human Computer Interaction* 16 (2001), Nr. 2-4, S. 97–166
- [DWC05] DAGGER, Declan ; WADE, Vincent ; CONLAN, Owen: Personalisation for All: Making Adaptive Course Composition Easy. In: *Educational Technology and Society* 8 (2005), Nr. 3, S. 9–25
- [ECH93] ELLIS, D. ; COX, D. ; HALL, K.: A comparison of the information seeking patterns of researchers in the physical and social sciences. In: *Journal of Documentation* 49 (1993), Nr. 4, S. 356–369
- [Ech07] ECHTER, Marcus: *Konzeption und Implementierung einer XML-basierten Repräsentation für Learning Maps im Rahmen des Wissenserstellungsprozesses bei SAP*, Universität Karlsruhe, Studienarbeit, 2007

- [ECTB97] ERSKINE, L. E. ; CARTER-TOD, D. R. N. ; ; BURTON, J. K.: Dialogical techniques for the design of web sites. In: *International Journal of Human Computer Studies* 47 (1997), S. 169–195
- [ED00] ERDMANN, Michael ; DECKER, Stefan: Ontology-aware XML Queries. In: *Proceedings of WebDB 2000*, 2000
- [ED03] ENGELBACH, Wolf ; DELP, Martin: Kontextbezogene Informationsversorgung: Anwenderanforderungen und Granularität der Modellierung. In: FÄHNRIICH, Klaus-Peter (Hrsg.) ; HERRE, Heinrich (Hrsg.): *Content- und Wissensmanagement. Beiträge auf den Leipziger Informatik-Tagen 2003 und Arbeiten aus dem Forschungsvorhaben PreBIS*, 2003
- [Efi04] EFIMOVA, Lilia: Discovering the iceberg of knowledge work: A weblog case. In: *Fifth European Conference on Organizational Knowledge, Learning and Capabilities (OKLC04)*, Innsbruck, 2004
- [EH97] ELLIS, D. ; HAUGAN, M.: Modelling the information-seeking patterns of engineers and research scientists in an industrial environment. In: *Journal of Documentation* 53 (1997), Nr. 4, S. 384–403
- [Ell89] ELLIS, D.: A behavioural model for information retrieval system design. In: *Journal of Information Science* 15 (1989), Nr. 4/5, S. 237–247
- [ER03] ERPENBECK, John ; ROSENSTIEL, Lutz von: *Handbuch Kompetenzmessung - Erkennen, verstehen und bewerten von Kompetenzen in der betrieblichen, pädagogischen und psychologischen Praxis*. Schäffer-Poeschel, 2003
- [Erp97] ERPENBECK, J.: Selbstgesteuertes, selbstorganisiertes Lernen. In: QUALIFIKATIONS-ENTWICKLUNGS-MANAGEMENT, Arbeitsgemeinschaft (Hrsg.): *Kompetenzentwicklung '97. Berufliche Weiterbildung in der Transformation – Fakten und Visionen*. Münster, New York, München, Berlin, 1997
- [Far03] FARMER, Johannes: Ad Hoc Coach System: Supporting Task-oriented Teaching and Learning under Time Pressure. In: RAUTERBERG, Matthias (Hrsg.) ; MENOZZI, Marino (Hrsg.) ; WESSON, Janet (Hrsg.): *Human-Computer Interaction INTERACT '03: IFIP TC13 International Conference on Human-Computer Interaction, 1st-5th September 2003, Zurich, Switzerland*, 2003. – ISBN 1–58603–363–8

- [FBW91] FAIX, Werner ; BUCHWALD, Christa ; WETZLER, Reiner: *Skill-Management – Qualifikationsplanung für Unternehmen und Mitarbeiter*. Wiesbaden : Gabler, 1991
- [FHH05] FIKES, Richard ; HAYES, Patrick ; HORROCKS, Ian: OWL-QL: A Language for Deductive Query Answering on the Semantic Web. In: *Journal on Web Semantics* 2 (2005), Nr. 1. <http://www.websemanticsjournal.org/ps/pub/2005-7>
- [FL85] FOLKMAN, Susan ; LAZARUS, Richard S.: If it Changes it Must be a Process: Study of Emotion and Coping During Three Stages of a College Examination. In: *Journal of Personality and Social Psychology* 49 (1985), Nr. 1, S. 150–170
- [FLDL04] FARMER, Johannes ; LINDSTAEDT, Stefanie ; DROSCHL, Georg ; LUTTENBERGER, P.: AD-HOC – Work-integrated Technology-supported Teaching and Learning. In: *5th International Conference on Organisational Knowledge, Learning, and Capabilities. Innsbruck, April 2-3 2004, 2004*
- [FR97] FUHR, Norbert ; RÖLLEKE, Thomas: A Probabilistic Relational Algebra for the Integration of Information Retrieval and Database Systems. In: *ACM Transactions on Information Systems* 15 (1997), Jan, Nr. 1, S. 32–66
- [FSD02] FREITAG, B. ; SÜSS, C. ; DZIARSTEK, C.: Adaption und Wiederverwendung von XML-basiertem eLearning-Content. In: *Informatik 2002, Workshop E-Learning Content auf Basis von XML*. Dortmund, 2002
- [Fuh95] FUHR, Norbert: Probabilistic Datalog - A Logic For Powerful Retrieval Methods. In: FOX, Edward A. (Hrsg.) ; INGWERSEN, Peter (Hrsg.) ; FIDEL, Raya (Hrsg.): *SIGIR'95, Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Seattle, Washington, USA, July 9-13, 1995 (Special Issue of the SIGIR Forum)*, ACM Press, 1995. – ISBN 0–89791–714–6, S. 282–290
- [Fuk05] FUKUSHIGE, Y.: Representing Probabilistic Relations in RDF. In: *International Semantic Web Conference (ISWC), Hiroshima, 2005*
- [GB01] GURP, J. van ; BOSCH, J.: Design, implementation and evolution of object oriented frameworks: concepts and guidelines. In: *Software - Practice and Experience* 31 (2001), S. 277–300
- [GBK06] GROSS, Tom ; BRAUN, Simone ; KRAUSE, Susanne: MatchBase: A Development Suite for Efficient Context-Aware Communication. In: *Fourteenth*

- Euromicro Conference on Parallel, Distributed, and Network-Based Processing - PDP 2006 (Feb. 15-17, Montbeliard-Sochaux, France)*. Los Alamitos, CA : IEEE Computer Society Press, 2006, S. 308–315
- [Ger97] GERDES, Heike: *Lernen mit Text und Hypertext*. Berlin : Pabst, 1997
- [GHJV96] GAMMA, E. ; HELM, R. ; JOHNSON, R. ; VLISSIDES, J.: *Entwurfsmuster: Elemente wiederverwendbarer objektorientierter Software*. Bonn : Addison-Wesley, 1996. – ISBN 3–89319–950–0
- [GHV05] GUTIÉRREZ, Claudio ; HURTADO, Carlos A. ; VAISMAN, Alejandro A.: Temporal RDF. In: *The Semantic Web: Research and Applications. Proceedings of the European Semantic Web Conference (ESWC 05)*, 2005, S. 93–107
- [GK03] GROSS, Tom ; KLEMKE, Roland: Context Modelling for Information Retrieval - Requirements and Approaches. In: *International Journal on WWW/Internet 1* (2003), Jun, Nr. 1, S. 29–42
- [GM95] GUPTA, Ashish ; MUMICK, Inderpal S.: Maintenance of Materialized Views: Problems, Techniques, and Applications. In: *IEEE Data Engineering Bulletin* 18 (1995), Nr. 2, S. 3–18
- [GM02] GÖKER, Ayse ; MYRHAUG, Hans I.: User Context and Personalisation. In: *ECCBR Workshop on Case Based Reasoning and Personalisation, Aberdeen*, 2002
- [Goe01a] GOESMANN, Thomas: KontextNavigator - Ein Organizational Memory zur Workflow-Unterstützung wissensintensiver Prozesse. In: *GI Jahrestagung (1)*, 2001, S. 61–67
- [Goe01b] GOESMANN, Thomas: KontextNavigator: A Workflow-Integrated Organizational Memory Information System to Support Knowledge-Intensive Processes. In: *Second International Workshop on Supporting Organisational Learning: KnowledgeManagement and Case-based Reasoning (SOL 01)*, 14th International Conference of Applications of Prolog (INAP2001), Tokio, 2001
- [GPT⁺05] GÜTL, Christian ; PIVEC, Maja ; TRUMMER, Christian ; GARCIA-BARRIOS, Victor M. ; MÖDRITSCHER, Felix ; PRIPFL, Jürgen ; UMGEHER, Martin: AdeLE (Adaptive e-Learning with Eye-Tracking): Theoretical Background, System Architecture and Application Scenarios. In: *European Journal of Open, Distance and E-Learning (EURODL)* 2005 (2005), Nr. II

- [Gua95] GUARINO, Nicola: Formal ontology, conceptual analysis and knowledge representation. In: *International Journal on Human-Computer Studies* 43 (1995), Nr. 5-6, S. 625–640
- [GWPZ04] GU, T. ; WANG, X.H. ; PUNG, H.K. ; ZHANG, D.Q.: An OWL-based Context Model in Intelligent Environments. In: *Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS'04)*. San Diego, California, 2004
- [Hal80] HALL, Stuart: Encoding/decoding. In: CONTEMPORARY CULTURAL STUDIES, Centre for (Hrsg.): *Culture, Media, Language: Working Papers in Cultural Studies 1972-1979*. London : Hutchinson, 1980, S. 128–138
- [Hap05] HAPKE, Thomas: 'In-formation' - Informationskompetenz und Lernen im Zeitalter Digitaler Bibliotheken. In: HAUKE, Petra (Hrsg.): *Bibliothekswissenschaft – quo vadis? = Library Science – quo vadis? : Eine Disziplin zwischen Traditionen und Visionen ; Programme – Modelle – Forschungsaufgaben*. Saur, 2005, S. 115–130
- [Has05] HASELOFF, Sandra: *Context Awareness in Information Logistics*, Technischen Universität Berlin, Diss., 2005
- [HBV06] HARZALLAH, Mounira ; BERIO, Giuseppe ; VERNADAT, François: Analysis and Modeling of Individual Competencies: Toward Better Management of Human Resources. In: *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 36 (2006), Nr. 1, S. 187–207
- [HCWA03] HOCKEMEYER, C. ; CONLAND, O. ; WADE, V. ; ALBERT, D.: Applying Competence Prerequisite Structures for eLearning and Skill Management. In: *Journal of Universal Computer Science* 9 (2003), Dezember, Nr. 12, S. 1428–1436
- [Hec03a] HECKMANN, Dominik: Introducing Situational Statements as an integrating Data Structure for User Modeling, Context-Awareness and Resource-Adaptive Computing. In: *Workshop Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen (LLWA03)*. Karlsruhe, 2003
- [Hec03b] HECKMANN, Dominik: A Specialized Representation for Ubiquitous Computing and User Modeling. In: *First Workshop on User Modeling for Ubiquitous Computing, UM 2003*, 2003

- [Hec06a] HECKMANN, Dominik: Situation Modeling and Smart Context Retrieval with Semantic Web Technology and Conflict Resolution. In: ROTHBERGHOFER, Thomas R. (Hrsg.) ; SCHULZ, Stefan (Hrsg.) ; LEAKE, David B. (Hrsg.): *Modeling and Retrieval of Context (MRC2005)* Bd. 3946, Springer-Verlag, Berlin Heidelberg, 2006 (LNAI), S. 34–47
- [Hec06b] HECKMANN, Dominik: *Ubiquitous User Modeling*. infix, 2006
- [Hen03] HENRICKSEN, Karen: *A framework for context-aware pervasive computing applications*, School of Information Technology and Electrical Engineering, University of Queensland, Diss., 2003
- [HH98] HOTH, J. ; HALL, W.: An Evaluation of Adapted Hypermedia Techniques Using Static User Modelling. In: *2nd Workshop on Adaptive Hypertext and Hypermedia of the Hypertext'98, Pittsburg, USA, 1998*
- [HI04] HENRICKSEN, Karen ; INDULSKA, Jadwiga: A Software Engineering Framework for Context-Aware Pervasive Computing. In: *PerCom[DBL04]*, S. 77–86
- [HI06] HENRICKSEN, Karen ; INDULSKA, Jadwiga: Developing context-aware pervasive computing applications: Models and approach. In: *Journal of Pervasive and Mobile Computing* 2 (2006), Nr. 1, S. 37–64
- [HIM05] HENRICKSEN, Karen ; INDULSKA, Jadwiga ; MCFADDEN, Terry: Middleware for Distributed Context-Aware Systems. In: *OnTheMove Federated Conferences 2004, International Symposium on Distributed Objects and Applications (DOA), Ayia Napa, Cyprus, Oct 31st-Nov 4th, 2005*, 2005, S. 846–863
- [HK03] HECKMANN, D. ; KRÜGER, A.: A User Modeling Markup Language (UserML) for Ubiquitous Computing. In: BRUSILOVSKY, P. (Hrsg.) ; CORBETT, A. (Hrsg.) ; ROSIS, F. de (Hrsg.): *Ninth International Conference on User Modeling (UM 2003)*, Springer, 2003
- [HKRS08] HITZLER, Pascal ; KRÖTZSCH, Markus ; RUDOLPH, Sebastian ; SURE, York: *Semantic Web Grundlagen*. Springer, 2008
- [Hoc03] HOCKEMEYER, Cord: Competence Based Adaptive E-Learning in Dynamic Domains. In: *oint Workshop of Cognition and Learning through Media–Communication for Advanced E-Learning (JWCL)*, 2003
- [HP05] HÄDRICH, Thomas ; PRIEBE, Torsten: Supporting Knowledge Work with Knowledge Stance-Oriented Integrative Portals. In: *Proceedings of the Thirteenth European Conference on Information Systems*, 2005

- [HR85] HAYES-ROTH, B.: A blackboard architecture for control. In: *Artificial Intelligence* 26 (1985), S. 251–321
- [Hru05] HRUBY, Pavel: Ontology-Based Domain-Driven Design. In: *Workshop on Best Practices for Model Driven Software Development, OOPSLA Conference*, 2005
- [HS07] HAPPEL, Hans-Jörg ; SCHMIDT, Andreas: Knowledge Maturing as a Process Model for Describing Software Reuse. In: GRONAU, Norbert (Hrsg.): *4th Conference Professional Knowledge Management - Experiences and Visions (WM '07), Potsdam Bd. 2*. Berlin : GITO-Verlag, 2007. – ISBN 978-3-936771-99-2, 155-164
- [HSB⁺05] HECKMANN, Dominik ; SCHWARTZ, Tim ; BRANDHERM, Boris ; SCHMITZ, Michael ; WILAMOWITZ-MOELLENDORFF, Margeritta von: GUMO – the General User Model Ontology. In: *10th International Conference on User Modeling (UM 05), Edinburgh*, 2005
- [Hug05] HUG, Theo: Microlearning and Narration. In: *MiT4: the work of stories - 4th Media in Transition Conference, Cambridge, MA, USA*, 2005
- [HZB⁺06] HEFKE, Mark ; ZACHARIAS, Valentin ; BIESALSKI, Ernst ; ABECKER, Andreas ; WANG, Qingli ; BREITER, Marco: An extendable Java Framework for Instance Similarities in Ontologies. In: *8th International Conference on Enterprise Information Systems, 23 - 27, May 2006, Paphos - Cyprus*, 2006
- [IB96] INGWERSEN, P. ; BORLUND, P.: Information transfer viewed as interactive cognitive processes. In: *Information Science: Integration in Perspective*. Copenhagen, Denmark : Royal School of Librarianship, 1996
- [IEE02] IEEE: Public and Private Information (PAPI) for Learner / IEEE. 2002 (IEEE P1484.2/D8). – Draft Standard for Learning Technology
- [IEE07] IEEE: IEEE 1484.20.1/Draft 7 Draft Standard for Learning Technology—Data Model for Reusable Competency Definitions / IEEE LTSC. 2007. – Forschungsbericht
- [IJ05] INGWERSEN, Peter ; JÄRVELIN, Kalervo: *The Turn: Integration of Information Seeking and Retrieval in Context*. Springer, 2005
- [IMS01] IMS: IMS Learner Information Package Specification (LIP) / IMS Global Learning Consortium Inc. 2001. – Forschungsbericht

- [ISO02] ISO: Topic Maps / ISO/IEC. 2002 (13250). – Forschungsbericht
- [Jam99] JAMESON, Anthony: User-Adaptive Systems: An Integrative Overview, 1999
- [Jam01] JAMESON, Anthony: Modeling Both the Context and the User. In: *Personal and Ubiquitous Computing* 5 (2001), Nr. 1, S. 29–33. – Available from <http://dfki.de/~jameson/abs/Jameson01PT.html>
- [JCGS92] JENSEN, C.S. ; CLIFFORD, J. ; GADIA, S.K. ; SNODGRASS, R.T.: A Glossary of Temporal Database Concepts. In: *ACM SIGMOD Record* 21 (1992), Nr. 3, S. 35–43
- [JI04] JÄRVELIN, Kalervo ; INGWERSEN, Peter: Information seeking research needs extension towards tasks and technology. In: *Information Research* 10 (2004), Nr. 1
- [Jon99] JONASSEN, D.: Designing Constructivist Learning Environments. In: REIGELUTH, C. M. (Hrsg.): *Instructional-design theories and models: a new paradigm of instructional theory*. 2nd edition. Mahwah, New Jersey : Lawrence Erlbaum Associates, 1999, S. 215–239
- [JS03] JUDD, Glenn ; STEENKISTE, Peter: Providing Contextual Information to Ubiquitous Computing Applications. In: *1st IEEE Conference on Pervasive Computing and Communication (PerCom 03), Fort Worth, 2003*, S. 133–142
- [JVM07] JARRAR, Mustafa ; VERVENNE, Luk ; MAYNARD, Diana: HR-Semantics Roadmap: The Semantic challenges and opportunities in the Human Resources domain / Ontology Outreach Advisory. 2007 (OOA-HR/2007-08-20/v025). – Forschungsbericht
- [KB02] KELLY, Diane ; BELKIN, Nicholas J.: Modeling Characteristics of the User's Problematic Situation with Information Search Use Behaviors. In: *Proceedings of the Second ACM/IEEE Joint Conference on Digital Libraries (JCDL 2002) Workshop on Document Search Interface Design for Large-scale Collections and Intelligent Access, 2002*
- [KGT06] KNÖPFEL, Andreas ; GRÖNE, Bernhard ; TABELING, Peter: *Fundamental Modeling Concepts*. Wiley, 2006
- [Kir94] KIRKPATRICK, Donald L.: *Evaluating Training Programs: The Four Levels*. San Francisco : Berrett-Koehler, 1994

-
- [Kir04] KIRCHHÖFER, Dieter: *Lernkultur Kompetenzentwicklung - Begriffliche Grundlagen*. Berlin, 2004
- [KKP01] KOBZA, Alfred ; KOENEMANN, J. ; POHL, W.: Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships. In: *The Knowledge Engineering Review* 16 (2001), Nr. 2, S. 111–155
- [Kle02] KLEMKE, R.: *Modelling Context in Information Brokering Processes*, RWTH Aachen, PhD, 2002
- [Kno03] KNOLMAYER, Gerhard F.: Decision Support Models for Composing and Navigating Through E-Learning Objects. In: *36th Annual Hawaii International Conference on System Sciences*, 2003
- [Kno04] KNOLMAYER, Gerhard F.: E-Learning Objects. In: *Wirtschaftsinformatik* 46 (2004), Nr. 3, S. 222–224
- [KO04] KAENAMPORN PAN, Manasawee ; O'NEILL, Eamonn: Modelling context: an Activity Theory approach. In: *2nd European Symposium on Ambient Intelligence, EUSAI 2004, Eindhoven, The Netherlands* Bd. 3295, Springer, 2004 (LNCS), S. 367–374
- [KO05] KAENAMPORN PAN, Manasawee ; O'NEILL, Eamonn: Integrating History and Activity Theory in Context Aware System Design. In: *1st International Workshop on Exploiting Context Histories in Smart Environments at Pervasive 2005, Munich*, 2005
- [Kor97] KOROSSY, Klaus: Extending the Theory of Knowledge Spaces: A Competence-Performance Approach. In: *Zeitschrift für Psychologie* 205 (1997), S. 53–82
- [Koy00] KOYCHEV, I.: Gradual Forgetting for Adaptation to Concept Drift. In: *ECAI 2000 Workshop on Current Issues in Spatio-Temporal Reasoning, Berlin, Germany*, 2000, S. 101–106
- [Kra99] KRAFT, S.: Selbstgesteuertes Lernen. In: *Zeitschrift für Pädagogik* 45 (1999), Nr. 6, S. 833–845
- [Kre00] KREUZ, Ingo: Considering the Dynamic in Knowledge Based Configuration. In: *14th Workshop New Results in Planning, Scheduling and Design, European Conference on Artificial Intelligence (ECAI 00)*, 2000

- [KS03] KURTZ, Cynthia F. ; SNOWDEN, David: The new dynamics of strategy: Sense-making in a complex and complicated world. In: *IBM Systems Journal* 42 (2003), Nr. 3, S. 462–483
- [KS04] KARAMPIPERIS, Pythagoras ; SAMPSON, Demetrios: Adaptive Instructional Planning Using Ontologies. In: *4th International Conference on Advanced Learning Technologies (ICALT 2004)*, 2004
- [KS05a] KARAMPIPERIS, P. ; SAMPSON, D.: Automatic Learning Object Selection and Sequencing in Web-Based Intelligent Learning Systems. In: MA, Zongmin (Hrsg.): *Web-Based Intelligent e-Learning Systems: Technologies and Applications*. Information Science Publishing, 2005, Kapitel 3, S. 56–71
- [KS05b] KARAMPIPERIS, Pythagoras ; SAMPSON, Demetrios: Adaptive Learning Resources Sequencing in Educational Hypermedia Systems. In: *Educational Technology & Society* 8 (2005), Nr. 4, S. 128–147
- [KS06] KUNZMANN, Christine ; SCHMIDT, Andreas: Ontology-based Competence Management for Healthcare Training Planning - A Case Study. In: *6th International Conference on Knowledge Management (I-KNOW 06)*, Graz, 2006
- [KS07] KUNZMANN, Christine ; SCHMIDT, Andreas: Kompetenzorientierte Personalentwicklung: Auf dem Wege zum Lernen bei Bedarf. In: *ERP Management 2007* (2007), S. 38–41
- [KSHS08] KUNZE, Christophe ; SCHMIDT, Andreas ; HOLTMANN, Carsten ; STORK, Wilhelm: Kontextsensitive Technologien und intelligente Sensorik für Ambient-Assisted-Living-Anwendungen. In: *1. Deutscher Kongress Ambient Assisted Living (AAL 2008)*, VDE Verlag, 2008
- [KSL07] KLEIN, Michael ; SCHMIDT, Andreas ; LAUER, Rolf: Ontology-Centred Design of an Ambient Middleware for Assisted Living: The Case of SOPRANO. In: KIRSTE, Thomas (Hrsg.) ; KÖNIG-RIES, Birgitta (Hrsg.) ; SALOMON, Ralf (Hrsg.): *Towards Ambient Intelligence: Methods in Cooperating Ensembles in Ubiquitous Environments (AIM-CU)*, Workshop at KI 2007, September 10, 2007, Osnabrück, Germany, 2007
- [KT05] KELLER, Tanja ; TERGAN, Sigmar-Olaf: *Knowledge and Information Visualization*. Springer, 2005
- [Kuh04] KUHLETHAU, Carol C.: *Seeking Meaning: A Process Approach to Library and Information Services*. 2nd edition. Westport, CT : Libraries Unlimited, 2004

- [Kun05] KUNZMANN, Christine: *Konzeption von Skills-Management-Instrumenten für die Bildungsbedarfsermittlung in der Pflege am Städtischen Klinikum Karlsruhe*, Hochschule Pforzheim, Diplomarbeit, 2005
- [Kuu96] KUUTTI, K.: Activity Theory as a Potential Framework for Human-Computer Interaction Research. In: NARDI, Bonnie A. (Hrsg.): *Context and Consciousness*. MIT Press, 1996, S. 17–44
- [LA03a] LEY, Tobias ; ALBERT, Dietrich: Identifying Employee Competencies in Dynamic Work Domains: Methodological Considerations and a Case Study. In: *Journal of Universal Computer Science* 9 (2003), S. 1500–1518
- [LA03b] LEY, Tobias ; ALBERT, Dietrich: Kompetenzmanagement als formalisierbare Abbildung von Wissen und Handeln für das Personalwesen. In: *Wirtschaftspsychologie* 2003 (2003), Nr. 3, S. 86–93
- [LB04] LONSDALE, Peter ; BEALE, Russell: Towards a dynamic process model of context. In: *Workshop on Advanced Context Modeling, Reasoning and Management, Ubicomp 2004*, 2004
- [LB05] LIN, Shin jeng ; BELKIN, Nicholas J.: Validation of a Model of Information Seeking Over Multiple Search Sessions. In: *Journal of the American Society for Information Science and Technology* 56 (2005), Nr. 4, S. 393–415
- [Lea02] LEARNING TECHNOLOGY STANDARDS COMMITTEE: IEEE Standard for Learning Object Metadata / IEEE Computer Society. 2002 (1484.12.1-2002). – Forschungsbericht
- [LF04] LINDSTAEDT, Stefanie N. ; FARMER, Johannes: Kooperatives Lernen in Organisationen. In: *CSCL-Kompendium. Lehr- und Handbuch zum computerunterstützten kooperativen Lernen*. Oldenbourg, 2004, S. 191–200
- [Lin01] LIN, Shin jeng: *Modeling and supporting multiple information seeking episodes*. New Brunswick, NJ, USA, Rutgers University, Diss., 2001
- [Lit04] LITTLER, Craig G.: *Perspectives on Learning and Information in Flexible Learning Environments*, University of Technology, Sydney, Diplomarbeit, 2004
- [LM06] LINDSTAEDT, Stefanie ; MAYER, Harald: A Storyboard of the APOSDLE Vision. In: *1st European Conference on Technology-Enhanced Learning (ECTEL 06)*, Crete, 2006

- [LTGF04] LEE, Shu-Shing ; THENG, Yin-Leng ; GOH, Dion Hoe-Lian ; FOO, Schubert Shou-Boon: Subjective Relevance: Implications on Digital Libraries for Experts and Novices. In: *Digital Libraries: International Collaboration and Cross-Fertilization*, Springer, 2004 (Lecture Notes in Computer Science), S. 453–457
- [LW91] LAVE, Jean ; WENGER, Etienne: *Situated Learning: Legitimate Peripheral Participation*. Cambridge, UK : Cambridge University Press, 1991
- [Mai04] MAIER, Ronald: *Knowledge Management Systems*. Berlin/Heidelberg : Springer, 2004
- [Mai05] MAIER, Ronald: Modeling Knowledge Work for the Design of Knowledge Infrastructures. In: *Journal of Universal Computer Science* 11 (2005), April, Nr. 4, S. 429–451
- [Mar95] MARCHIONINI, G. ; ANONYMOUS (Hrsg.): *Information Seeking in Electronic Environments*. Cambridge University, 1995
- [MCF⁺94] MITCHELL, T. ; CARUNA, R. ; FREITAG, D. ; MCDERMOTT, J. ; ZABOWSKI, D.: Experience with a Learning Personal Assistant. In: *Communications of the ACM* 37 (1994), Nr. 7, S. 81–91
- [McK99] MCKENZIE, Jamie: Scaffolding for Success. In: *From Now On - The Educational Technology Journal* 9 (1999), Nr. 4
- [Mea75] MEAD, George H.: *Geist, Identität, Gesellschaft – aus Sicht des Sozialbehaviorismus*. Suhrkamp, 1975
- [MEJ⁺06] MAGERKURTH, C. ; ETTER, R. ; JANSE, M. ; KELA, J. ; KOCSIS, O. ; RAMPARANY, F.: An Intelligent User Service Architecture for Networked Home Environments. In: *2nd International Conference on Intelligent Environments, Athens, Greece, July 5-6, 2006*, 2006, S. 361–370
- [Men00] MENTZAS, Gregoris N.: The two faces of knowledge management. In: *International Consultant's Guide 2000* (2000), May, S. 10–11
- [Mer74] MERTENS, Dieter: Schlüsselqualifikationen. Thesen zur Schulung für eine moderne Gesellschaft. In: *Mitteilungen aus der Arbeitsmarkt- und Berufsforschung* 7 (1974)
- [MG04] MIKA, Peter ; GANGEMI, Aldo: Descriptions of Social Relations. In: *First International Workshop on FOAF, Social Networks and the Semantic Web, Galway, Ireland, 2004*

- [MGBG04] MÖDRITSCHER, Felix ; GARCIA-BARRIOS, Victor M. ; GÜTL, Christian: The Past, the Present and the Future of adaptive E-Learning. In: *Proceedings of ICL, Villach, Austria, 2004*
- [MH68] MCCARTHY, J. ; HAYES, P. J.: *Some philosophical problems from the standpoint of artificial intelligence*. Report Memo AI-73, Department of Computer Science, Stanford University, Stanford, California, 1968
- [MHM⁺04] MATSUO, Yutaka ; HAMASAKI, Masahiro ; MORI, Junichiro ; TAKEDA, Hideaki ; HASIDA, Koiti: Ontological Consideration on Human Relationship Vocabulary for FOAF. In: *Proceedings of the 1st Workshop on Friend of a Friend, Social Networking and Semantic Web, 2004*
- [MHP05] MAIER, Ronald ; HÄDRICH, Thomas ; PEINL, René: *Enterprise Knowledge Infrastructures*. Springer, 2005
- [Mit02] MITCHELL, Keith: *Supporting the Development of Mobile Context-Aware Computing*, Lancaster University, Diss., 2002
- [MMS03] MAEDCHE, Alexander ; MOTIK, Boris ; STOJANOVIC, Ljiljana: Managing Multiple and Distributed Ontologies in the Semantic Web. In: *VLDB Journal* 12 (2003), Nr. 4, S. 286–302
- [Mos05] MOSEL, Stephan: *Praktiken selbstgesteuerten Lernens anhand der Nutzung von web-basierten Personal-Publishing-Systemen*, Justus Liebig Universität Gießen, Institut für Erziehungswissenschaft, Diplomarbeit, 2005
- [Mot94] MOTRO, Amihai: Management of Uncertainty in Database Systems. In: KIM, W. (Hrsg.): *Modern Database Systems: the Object Model, Interoperability and Beyond*. Addison-Wesley / ACM Press, 1994, S. 457–476
- [Mot06] MOTIK, Boris: *Reasoning in Description Logics using Resolution and Deductive Databases*, University of Karlsruhe, Diss., 2006
- [MPT07] MORBIDONI, Christian ; POLLERES, Axel ; TUMMARELLO, Giovanni: Who the FOAF knows Alice? RDF Revocation in DBin 2.0. In: *4th Italian Semantic Web Workshop SEMANTIC WEB APPLICATIONS AND PERSPECTIVES (SWAP), Bari, Italy, December 2007, 2007*
- [MS95] MARCH, S. ; SMITH, G.: Design and Natural Science Research on Information Technology. In: *Decision Support Systems* 15 (1995), S. 251–266

- [MS02a] MAGLAUGHLIN, Kelly L. ; SONNENWALD, Diane H.: User Perspectives on Relevance Criteria: A Comparison among Relevant, Partially Relevant, and Not-Relevant Judgments. In: *Journal of the American Society for Information Science and Technology* 53 (2002), Nr. 5, S. 327–342
- [MS02b] MAYR, Peter ; SEUFERT, Sabine: *Fachlexikon e-learning*. managerSeminare, 2002
- [MS07] MAIER, Ronald ; SCHMIDT, Andreas: Characterizing Knowledge Maturing: A Conceptual Process Model for Integrating E-Learning and Knowledge Management. In: GRONAU, Norbert (Hrsg.): *4th Conference Professional Knowledge Management - Experiences and Visions (WM '07)*, Potsdam Bd. 2. Berlin : GITO-Verlag, 2007. – ISBN 978–3–936771–99–2, 325-334
- [MVB⁺04] MASOLO, Claudio ; VIEU, Laure ; BOTTAZZI, Emanuele ; CATENACCI, Carola ; FERRARIO, Roberta ; GANGEMI, Aldo ; GUARINO, Nicola: Social Roles and their Descriptions. In: *Proceedings of the Ninth International Conference on the Principles of Knowledge Representation and Reasoning (KR2004)*, Whistler, Canada, June 2-5, 2004, 2004
- [NF04] NOTTELMANN, H. ; FUHR, N.: pDAML+OIL: A Probabilistic Extension to DAML+OIL Based on Probabilistic Datalog. In: *Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Perugia, Italy, 2004
- [Nie03] NIEDZWIEDZKA, Barbara: A proposed general model of information behaviour. In: *Information Research* 9 (2003), Nr. 1. <http://InformationR.net/ir/9-1/paper164.html>
- [Nis02] NISSEN, Mark E.: An Extended Model of Knowledge Flow Dynamics. In: *Communications of the Associations for Information Systems* 8 (2002), S. 251–266
- [NL02] NISSEN, Mark ; LEVITT, Raymond: Dynamic Models of Knowledge Flow Dynamics / Stanford University. 2002 (76). – CIFE Working Paper
- [Non94] NONAKA, Ikujiro: A Dynamic Theory of Organizational Knowledge Creation. In: *Organization Science* 5 (1994), Nr. 1, S. 14–37
- [NT95] NONAKA, Ikujiro ; TAKEUCHI, Hirotaka: *The Knowledge-Creating Company*. Oxford University Press, 1995
- [OBCW06] O'KEEFFE, Ian ; BRADY, Aoife ; CONLAN, Owen ; WADE, Vincent: Just-in-Time Generation of Pedagogically Sound, Context Sensitive Personalized

- Learning Experiences. In: *International Journal on E-Learning* 5 (2006), Nr. 1, S. 113–127
- [OJ93] O'DAY, V.L. ; JEFFRIES, R.: Orienteering in an Information Landscape: How Information Seekers Get From Here to There. In: *INTERCHI 93, Amsterdam, The Netherlands, 1993*
- [Ost05] OSTYN, Claude: General architecture for a SCORM 2004 LMS implementation / Ostyn Consulting. Version: 2005. <http://ostyn.com/standards/docs/GeneralArchitectureForLMSImplementation-1.htm>. 2005. – Forschungsbericht
- [Ost06] OSTYN, Claude: Service Oriented Architecture for Competency-based Lifelong Learning and Personal Development / Ostyn Consulting. 2006. – Forschungsbericht. – 11. Oktober 2006
- [Ove01] OVERWIEN, Bernd: Debatten, Begriffsbestimmungen und Forschungsansätze zum informellen Lernen und zum Erfahrungslernen. In: ARBEIT, Soziales und F. f. (Hrsg.): *Tagungsband zum Kongreß Der flexible Mensch*. Berlin : BBJ-Verlag, 2001, S. 359–376
- [Par00] PARSONS, T.: *Indeterminate identity. Metaphysics and semantics*. Clarendon Press, 2000
- [PB98] PROBST, G. ; BÜCHEL, B.: *Organisationales Lernen - Wettbewerbsvorteil der Zukunft*. Gabler, 1998
- [PF05] PICOT, Arnold (Hrsg.) ; FISCHER, Tim (Hrsg.): *Weblogs. Grundlagen, Konzepte und Praxis im unternehmerischen Umfeld*. dpunkt, 2005
- [PH98] PARSONS, Simon ; HUNTER, Anthony: A review of uncertainty handling formalisms. In: PARSONS, Simon (Hrsg.) ; HUNTER, Anthony (Hrsg.): *Applications of uncertainty formalisms*. New York : Springer, 1998 (Lecture Notes in Computer Science 1455), S. 8–37
- [PL04] PARK, Ok-Choon ; LEE, Jung: Adaptive Instructional Systems. In: JONASSEN, David H. (Hrsg.): *Handbook of research for educational communications and technology*. 2nd edition. Mahwah, N.J. : Lawrence Erlbaum, 2004, Kapitel 25, S. 651–684
- [PLO⁺04] POWER, Ruaidhri ; LEWIS, Dave ; O'SULLIVAN, Declan ; CONLAN, Owen ; WADE, Vincent: A Context Information Service using Ontology-Based

- Queries. In: *Sixth International Conference on Ubiquitous Computing (Ubi-comp'04)*, 2004
- [PNS⁺00] PETRELLI, D. ; NOT, E. ; STRAPPARAVA, C. ; STOCK, O. ; ZANCANARO, M.: Modeling Context is Like Taking Pictures. In: *Conference on Human Factors in Computers, Workshop The What, Who, Where, When, Why and How of Context-Awareness*", 2000
- [Pol67] POLANYI, Michael: *The implicit dimension*. Anchor Books, 1967
- [Pow03] POWER, R.: Topic Maps for Context Management. In: *Workshop on Adaptive Systems for Ubiquitous Computing*, 2003
- [Pre97] PREE, Wolfgang: *Komponentenbasierte Softwareentwicklung mit Frameworks*. dpunkt, 1997
- [PRR03] PROBST, Gilbert ; RAUB, Stefan ; ROMHARDT, Kai: *Wissen managen: Wie Unternehmen ihre wertvollste Ressource optimal nutzen*. Wiesbaden : Gabler, 2003
- [PS06] PRUD'HOMMEAUX, Eric ; SEABORNE, Andy: SPARQL Query Language for RDF / W3C. Version: 2006. <http://www.w3.org/TR/rdf-sparql-query/>. 2006. – W3C Working Draft
- [Ray03] RAY, Judith M.: *Designing a Knowledge Management System: A Sensemaking Perspective*, Pennsylvania State University, Diss., 2003
- [Röd03] RÖDER, Stefan: *Eine Architektur für individualisierte computergestützte Lernumgebungen*. Frankfurt/Main : Lang, 2003
- [Rei05] REINMANN, Gabi: Wissensmanagement und Medienbildung – neue Spannungsverhältnisse und Herausforderungen. In: *MedienPädagogik* 05 (2005)
- [Rem02] REMUS, Ulrich: *Prozessorientiertes Wissensmanagement - Konzepte und Modellierung*, Universität Regensburg, Diss., 2002
- [Ren96] RENKL, Alexander: Träges Wissen: Wenn Erlerntes nicht genutzt wird. In: *Psychologische Rundschau* 47 (1996), Nr. 2, S. 78–92
- [Rey01] REYNOLDS, Carson J.: *The Sensing and Measurement of Frustration with Computers*, Massachusetts Institute of Technology, Diss., 2001
- [Rho00] RHODES, B.J.: *Just-In-Time Information Retrieval*, MIT, Diss., 2000

- [RL02] RUTHVEN, Ian ; LALMAS, Mounia: Using Dempster-Shafer's Theory of Evidence to combine aspects of information use. In: *Journal of Intelligent Systems* 19 (2002), Nr. 2, S. 267–302
- [Röl03] RÖLL, Franz-Josef: *Pädagogik der Navigation - Selbstgesteuertes Lernen durch Neue Medien*. Kopäd, 2003
- [Röl05] RÖLL, Martin: Corporate E-Learning mit Weblogs und RSS. In: HOHENSTEIN, Andreas (Hrsg.) ; WILBERS, Karl (Hrsg.): *Handbuch E-Learning. Expertenwissen aus Wissenschaft und Praxis*. Ergänzungslieferung April 2005. Köln : Deutscher Wirtschaftsdienst, 2005
- [RLFVDRPA06] REY-LÓPEZ, Marta ; FERNÁNDEZ-VILAS, Ana ; DÍAZ-REDONDO, Rebeca P. ; PAZOS-ARIAS, José J.: Providing SCORM with Adaptivity. In: *World-Wide Web Conference (WWW '06), Edinburgh, Scotland, 2006*
- [RM00] RHODES, B.J. ; MAES, P.: Just-in-time information retrieval agents. In: *IBM Systems Journal* 39 (2000), Nr. 3/4, S. 685–704
- [RN03] RUSSELL, Stuart ; NORVIG, Peter: *Artificial Intelligence: A Modern Approach*. 2nd Edition. Prentice Hall, 2003
- [Rot02] ROTHWELL, William J.: *The Workplace Learner*. American Management Association, 2002
- [RPM97] RYAN, N. ; PASCOE, J. ; MORSE, D.: Enhanced reality fieldwork: the context-aware archaeological assistant. In: GAFFNEY, V. (Hrsg.) ; LEUSEN, M. van (Hrsg.) ; EXXON, S. (Hrsg.): *Computer Applications and Quantitative Methods in Archaeology (CAA 97), Oxford, 1997*
- [RPS06] RAMASUBRAMANIAN, Venugopalan ; PETERSON, Ryan ; SIRER, Emin G.: Corona: A High Performance Publish-Subscribe System for the World Wide Web. In: *Proceedings of Networked System Design and Implementation (NSD 06), 2006*
- [RR01] REINMANN-ROTHMEIER, Gabi: Wissen managen: Das Münchener Modell / Lehrstuhl für Empirische Pädagogik und Pädagogische Psychologie, LMU München. 2001 (131). – Forschungsbericht
- [RS06] ROSTANIN, Oleg ; SCHIRRU, Rafael: Identification of User's Learning Goals in Workflows. In: *oint International Workshop on Professional Learning, Competence Development and Knowledge Management - LOKMOL and L3NCD, in conjunction with the 1st European Conference on Technology Enhanced Learning, 2006*

- [Rya99] RYAN, Nick: ConteXtML: Exchanging Contextual Information between a Mobile Client and the FieldNote Server, 1999
- [Sar96] SARACEVIC, Tefko: Relevance reconsidered. In: *Information science: Integration in perspectives. Proceedings of the Second Conference on Conceptions of Library and Information Science, Copenhagen (Denmark)*, 1996
- [Sat01] SATYANARAYANAN, M.: Pervasive Computing: Vision and Challenges. In: *IEEE Personal Communications Magazine* 2001 (2001), S. 10–17
- [SAW94] SCHILIT, B. ; ADAMS, N. ; WANT, R.: Context-Aware Computing Applications. In: *IEEE Workshop on Mobile Computing Systems and Applications*, 1994, S. 85–90
- [SB06] SCHMIDT, Andreas ; BRAUN, Simone: Context-Aware Workplace Learning Support: Concept, Experiences, and Remaining Challenges. In: NEJDL, Wolfgang (Hrsg.) ; TOCHTERMANN, Klaus (Hrsg.): *Innovative Approaches for Learning and Knowledge Sharing. First European Conference on Technology-Enhanced Learning (EC-TEL 2006), Crete, Greece. Proceedings*. Bd. 4227, Springer, 2006 (Lecture Notes in Computer Science). – ISBN 978–3–540–45777–0, 518–524
- [SBG98] SCHMIDT, Albrecht ; BEIGL, Michael ; GELLERSEN, Hans W.: There is more to Context than Location. In: *International workshop of Interactive Applications of Mobile Computing*. Rostock, 1998
- [Sch95] SCHILIT, William N.: *A System Architecture for Context-Aware Computing*, Columbia University, Diss., 1995
- [Sch02a] SCHEIR, Peter: *Wissensmanagement zur Unterstützung von Kundenbeziehungsmanagement*, Universität Graz, Magisterarbeit, 2002
- [Sch02b] SCHMIDT, Albrecht: *Ubiquitous Computing - Computing in Context*, University of Lancaster, PhD, 2002
- [Sch03a] SCHMITT, Bethina: *Benutzerprofile für die Anfrageverarbeitung in verteilten Digitalen Bibliotheken*, Universität Karlsruhe, Fakultät für Informatik, Diss., 2003
- [Sch03b] SCHÜTT, Peter: Wissensmanagement im eLearning. In: *SCIL Congress 2003: Shaping Innovations - eLearning as a Catalyst for a New Teaching and Learning Culture?*, St. Gallen,, 2003

- [Sch04a] SCHMIDT, Andreas: Context-Steered Learning: The Learning in Process approach. In: *IEEE International Conference on Advanced Learning Technologies (ICALT '04)*. Joensuu, Finland : IEEE Computer Society, Sep 2004, 684-686
- [Sch04b] SCHMIDT, Andreas: Kontextgesteuertes Lernen in Unternehmensumgebungen: Der Learning in Process-Ansatz. In: *2. Deutsche e-Learning Fachtagung der Gesellschaft für Informatik (Delfi 04)*, Paderborn, Germany. Bonn : Gesellschaft für Informatik, Sep 2004 (Lecture Notes in Informatics), 259-270
- [Sch05a] SCHMIDT, Andreas: Bridging the Gap Between Knowledge Management and E-Learning with Context-Aware Corporate Learning Solutions. In: ALTHOFF, Klaus-Dieter (Hrsg.) ; DENGEL, Andreas (Hrsg.) ; BERGMANN, Ralph (Hrsg.) ; NICK, Markus (Hrsg.) ; ROTH-BERGHOFER, Thomas (Hrsg.): *Professional Knowledge Management. Third Biennial Conference, WM 2005, Kaiserlautern, Germany, April 2005. Revised Selected Papers* Bd. 3782, Springer, 2005 (Lecture Notes in Artificial Intelligence), 203–213
- [Sch05b] SCHWARZ, Sven: A Context Model for Personal Knowledge Management. In: *Proceedings of the IJCAI-05 Workshop on Modeling and Retrieval of Context Edinburgh, July 31 - August 1, 2005*, 2005 (CEUR Workshop Proceedings)
- [SE05] SEUFERT, Sabine ; EULER, Dieter: Learning Design: Gestaltung eLearning-gestützter Lernumgebungen in Hochschulen und Unternehmen / Swiss Centre for Innovations in Learning - Institut für Wirtschaftspädagogik, Universität St. Gallen. 2005 (5). – SCIL-Arbeitsbericht
- [See03] SEEBERG, C.: *Life Long Learning; Modulare Wissensbasen für elektronische Lernumgebungen*. Springer, 2003
- [SF02] SENTZ, Karl ; FERSON, Scott: Combination of Evidence in Dempster-Shafer Theory / Sandia National Laboratories. Albuquerque, New Mexico, USA, 2002 (SAND2002-0835). – SANDIA REPORT
- [SG01] SCHMIDT, Albrecht ; GELLERSEN, Hans-Werner: Modell, Architektur und Plattform für Informationssysteme mit Kontextbezug. In: *Informatik Forschung und Entwicklung* 16 (2001), Nr. 4, S. 213–224
- [SG03] SCHREYÖGG, Georg ; GEIGER, Daniel: Kann die Wissensspirale Grundlage des Wissensmanagements sein? / Freie Universität Berlin, Fachbereich Wirtschaftswissenschaft, Institut für Management. 2003 (20/03). – Diskussionsbeiträge des Instituts für Management

- [Shn98] SHNEIDERMAN, Ben: Relate – create – donate: a teaching/learning philosophy for the cyber-generation. In: *Computers & Education* 31 (1998), Nr. 1, 25–39. <ftp://ftp.cs.umd.edu/pub/hcil/Reports-Abstracts-Bibliography/97-17html/97-17.html>
- [Shn02] SHNEIDERMAN, Ben: *Leonardo's Laptop: Human Needs and the new Computing Technologies*. Cambridge, Mass., 2002
- [Sie05] SIEMENS, George: *Connectivism: Learning as Network-Creation*. elearn-space. <http://www.elearnspace.org/Articles/networks.htm>. Version: Aug 2005
- [SKB06] SCHMIDT, Andreas ; KUNZMANN, Christine ; BIESALSKI, Ernst: Systematische Personalentwicklung mit ontologiebasierten Kompetenzkatalogen. In: GRONAU, Norbert (Hrsg.) ; FRÖMING, Jane (Hrsg.) ; SCHMID, Simone (Hrsg.): *Fachtagung Kompetenzmanagement, GITO, 2006*
- [SLP04] STRANG, Thomas ; LINNHOF-POPIEN, Claudia: A Context Modeling Survey. In: *Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004 - The Sixth International Conference on Ubiquitous Computing, Nottingham/England, 2004*
- [SLPF03] STRANG, Thomas ; LINNHOF-POPIEN, Claudia ; FRANK, Korbinian: CoOL: A Context Ontology Language to enable Contextual Interoperability. In: STEFANI, Jean-Bernard (Hrsg.) ; DAMEURE, Isabelle (Hrsg.) ; HAGIMONT, Daniel (Hrsg.): *LNCS 2893: Proceedings of 4th IFIP WG 6.1 International Conference on Distributed Applications and Interoperable Systems (DAIS2003)* Bd. 2893. Paris/France : Springer Verlag, November 2003 (Lecture Notes in Computer Science (LNCS)), 236–247
- [SN97] SCHREYÖGG, G. ; NOSS, Chr.: Zur Bedeutung des organisationalen Wissens für organisatorische Lernprozesse. In: *Handbuch Lernende Organisation. Unternehmens- und Mitarbeiterpotentiale erfolgreich erschließen*. Gabler, 1997
- [Sno00] SNOWDEN, David: The Social Ecology of Knowledge Management. In: CHAUVEL, Daniel (Hrsg.) ; DESPRES, Charles (Hrsg.): *Knowledge Horizons*. Boston : Butterworth-Heinemann, 2000, Kapitel 12, S. 237–266
- [Sno02] SNOWDEN, David: Complex acts of knowing – paradox and descriptive self-awareness. In: *Journal of Knowledge Management* 6 (2002), Nr. 2

- [SR02] SCHIERSMANN, Christiane ; REMMELE, Heide: Neue Lernarrangements in Betrieben: Theoretische Fundierung – Einsatzfelder – Verbreitung. In: *QUEM-report 75* (2002)
- [SR04] SEILER, Thomas B. ; REINMANN, Gabi: Der Wissensbegriff im Wissensmanagement: Eine strukturgenetische Sicht. In: REINMANN, Gabi (Hrsg.) ; MANDL, Heinz (Hrsg.): *Psychologie des Wissensmanagements: Perspektiven, Theorien und Methoden*. Hogrefe, 2004
- [SSÖ03] SNIS, Ulrika L. ; SVENSSON, Lars ; ÖSTLUND, Christian: Towards work-integrated learning communities. In: *Proceedings of the 11th European Conference on Information Systems, ECIS 2003, Naples, Italy 16-21 June 2003*, 2003
- [SSR04] SONNTAG, Karlheinz ; SCHMIDT-RATHJENS, Claudia: Kompetenzmodelle - Erfolgsfaktoren im HR-Management? Ein strategie- und evidenzbasierter Ansatz der Kompetenzmodellierung. In: *Personalführung 37* (2004), Nr. 10, S. 18–26
- [Sta01] STACY, R.: *Complex responsive processes in organizations: learning and knowledge creation*. Routledge, 2001
- [Ste00] STEGMAIER, Ralf: *Kompetenzentwicklung durch arbeitsintegriertes Lernen in der Berufsbildung*, Fakultät für Verhaltens- und Sozialwissenschaften, Universität Heidelberg, Diss., 2000
- [Ste02] STENMARK, Dick: Information vs. Knowledge: The Role of intranets in Knowledge Management. In: *Proceedings of the 35th Hawaii International Conference on System Sciences*, IEEE Press, 2002
- [Str03] STRANG, Thomas: *Service Interoperability in Ubiquitous Computing Environments*, Ludwig-Maximilians-University Munich, Diss., Oktober 2003
- [Str04] STRAKA, Gerald A.: Informal Learning: genealogy, concepts, antagonisms and questions / University of Bremen, Germany. Version: Nov 2004. http://www.itb.uni-bremen.de/downloads/Publikationen/Forschungsberichte/fb_15_04.pdf. 2004 (15/2004). – ITB Forschungsberichte
- [Str05] STRAUB, Richard: *Lernen im Kontext - Dynamischen Lernkonzepten gehört die Zukunft*. Competence-Site. [http://www.competence-site.de/elearning.nsf/0/469f61af7e05d337c1256f95004b4b4c?](http://www.competence-site.de/elearning.nsf/0/469f61af7e05d337c1256f95004b4b4c?OpenDocument) OpenDocument. Version: Sep 2005

- [SW03] SCHMIDT, Andreas ; WINTERHALTER, Claudia: User Context Aware Delivery of E-Learning Material: Approach and Architecture. In: *I-KNOW 2003*. Graz, Austria, 2003
- [SW04] SCHMIDT, Andreas ; WINTERHALTER, Claudia: User Context Aware Delivery of E-Learning Material: Approach and Architecture. In: *Journal of Universal Computer Science (JUCS)* 10 (2004), Nr. 1, 28–36. http://www.andreas-p-schmidt.de/publications/jucs04_schmidt_context-aware-delivery.pdf
- [Swa86] SWANSON, D.R.: Subjective versus Objective Relevance in Bibliographic Retrieval Systems. In: *Library Quarterly* 56 (1986), Oct, Nr. 4, S. 389–398
- [Swe93] SWELLER, J.: Some cognitive processes and their consequences for the organisation and presentation of information. In: *Australian Journal of Psychology* 45 (1993), Nr. 1, S. 1–8
- [Swe94] SWELLER, J.: Cognitive load theory, learning difficulty and instructional design. In: *Learning and Instruction* 4 (1994), S. 295–312
- [SZ95] STEIN, E.W. ; ZWASS, V.: Actualizing Organizational Memory with Information Systems. In: *Information Systems Research* 6 (1995), Nr. 2, S. 85–117
- [TA01] TEKINERDOGAN, B. ; AKSIT, M.: Classifying and Evaluating Architecture Design Methods. In: AKSIT, M. (Hrsg.): *Software Architectures and Component Technology: The State of the Art in Research and Practice*. Kluwer Academic Publishers, 2001, S. 3–27
- [Tay68] TAYLOR, Robert S.: The process of asking questions. In: *American Documentation* 13 (1968), Oct, Nr. 4, S. 391–396
- [TGNO92] TERRY, Douglas ; GOLDBERG, David ; NICHOLS, David ; OKI, Brian: Continuous queries over append-only databases. In: *SIGMOD '92: Proceedings of the 1992 ACM SIGMOD international conference on Management of data*. New York, NY, USA : ACM Press, 1992. – ISBN 0–89791–521–6, S. 321–330
- [TLFH05] TOSH, David ; LIGHT, Tracy P. ; FLEMING, Kele ; HAYWOOD, Jeff: Engagement with Electronic Portfolios: Challenges from the Student Perspective. In: *Canadian Journal of Learning and Technology* 31 (2005), Nr. 2
- [TOKF05] TOMS, Elaine G. ; O'BRIEN, Heather L. ; KOPACK, Rick ; FREUND, Luanne: Searching for Relevance in the Relevance of Search. In: *Information Context: Nature, Impact, and Role* Bd. 3507, Springer, 2005 (Lecture Notes in Computer Science), S. 59–78

- [TS04] TERGAN, S. O. ; SCHENKEL, P.: *Was macht E-Learning erfolgreich? Grundlagen und Instrumente der Qualitätsbeurteilung*. Springer, 2004
- [TSM02] THORNE, Scott ; SHUBERT, C. ; MERRIMAN, J.: OKI Architecture Overview / MIT. 2002. – Forschungsbericht
- [TT91] TANAKA, J.W. ; TAYLOR, M.: Object categories and expertise: Is the basic-level in the eye of the beholder? In: *Cognitive Psychology* 23 (1991), S. 457–482
- [Tuo99] TUOMI, Ikka: Data is More than Knowledge: Implications of the Reversed Knowledge Hierarchy for Knowledge Management and Organizational Memory. In: *Journal of Management Information Systems* 16 (1999), Nr. 3, S. 103–117
- [TVTY90] TAKEDA, H. ; VEERKAMP, P. ; TOMIYAMA, T. ; YOSHIKAWAM, H.: Modeling Design Processes. In: *AI Magazine Winter* (1990), S. 37–48
- [TW04] TOSH, David ; WERDMULLER, Ben: The Learning Landscape: a conceptual framework for ePortfolios. In: *interact 2004* (2004), Nr. 29, S. 14–15
- [UG96] USCHOLD, Mike ; GRÜNINGER, Michael: Ontologies: Principles, Methods and Applications. In: *Knowledge Engineering Review* 11 (1996), Nr. 2, S. 93–155
- [Uhl06] UHL, Michael: *Konzeption und Implementierung einer Erweiterung des SCORM-Standards für kontextbewusste Lernobjekte*, Universität Karlsruhe, Diplomarbeit, 2006
- [Ull97] ULLMANN, Jeffrey: Information Integration using Logical Views. In: *6th International Conference on Database Theory*, Springer, 1997, S. 19–40
- [USLG06] ULBRICH, Armin ; SCHEIR, Peter ; LINDSTAEDT, Stefanie N. ; GÖRTZ, Manuel: A Context-Model for Supporting Work-Integrated Learning. In: *Innovative Approaches for Learning and Knowledge Sharing. Proceedings of the First European Conference on Technology-Enhanced Learning (ECTEL 06)*, 2006, S. 525–530
- [USM06] UDREA, O. ; SUBRAHMANIAN, V.S. ; MAJKIC, Z.: Probabilistic RDF. In: *IEEE International Conference on Information Reuse and Integration*, 2006, S. 172–177
- [Vaz01] VAZIRANI, V. V.: *Approximation Algorithms*. Springer, 2001

- [WB97] WANG BALDONADO, Michelle Q.: *An Interactive, Structure-Mediated Approach to Exploring Information in a Heterogeneous, Distributed Environment*, Department of Computer Science of Stanford University, Diss., Dec 1997
- [WBR04] WILSON, Scott ; BLINCO, Kerry ; REHAK, Daniel: *An e-Learning Framework: A Summary / JISC-CETIS*. Version:2004. <http://lsal.org/lsal/expertise/papers/notes/serviceframeworks/elf20040720/elfsummary-20040720.pdf>. 2004. – Forschungsbericht
- [Web98] WEBER, Stefan: *Investigations in Belnap's Logic of Inconsistent and Unknown Information*, Universität Leipzig, Diss., 1998
- [WGZP04] WANG, X. ; GU, T. ; ZHANG, D. ; PUNG, H.: *Ontology Based Context Modeling and Reasoning using OWL*. In: *IEEE International Conference on Pervasive Computing and Communication (PerCom'04), Orlando, Florida, 2004*
- [Wil73] WILSON, P.: *Situational Relevance*. In: *Information Storage and Retrieval* 9 (1973), Nr. 8, S. 457–471
- [Wil81] WILSON, T. D.: *On User Studies and Information Needs*. In: *Journal of Librarianship* 37 (1981), Nr. 1, 3–15. <http://informationr.net/tdw/publ/papers/1981infoneeds.html>
- [Wil99] WILSON, T. D.: *Models in Information Behaviour Research*. In: *Journal of Documentation* 55 (1999), Jun, Nr. 3, S. 249–270
- [Wil00] WILEY, David: *Connecting Learning Objects to Instructional Design Theory: A Definition, a Metaphore, and a Taxonomy*. In: *The Instructional Use of Learning Objects*. University of Utah, 2000
- [Wil01] WILLKE, Helmut: *Systemisches Wissensmanagement*. 2. neubearbeitete Auflage. Lucius und Lucius, 2001
- [Wil03] WILBERS, Karl: *E-Learning*. In: *Berufsbildung* 57 (2003), Nr. 80
- [Win02] WINTERHALTER, Claudia: *Kontextbasierte Unterstützung von arbeitsplatzgebundenem Lernen in Unternehmensumgebungen*, Fakultät für Informatik, Universität Karlsruhe, Diplomarbeit, 2002
- [WK96a] WEBB, Geoffrey I. ; KUZMYCZ, Mark: *Feature Based Modelling: A Methodology for Producing Coherent, Consistent, Dynamically Changing Models of Agents Competencies*. In: *User Modelling and User-Adapted Interaction* 5 (1996), Nr. 2, S. 117–150

- [WK96b] WIDMER, G. ; KUBAT, M.: Learning in the presence of concept drift and hidden contexts. In: *Machine Learning* 23 (1996), S. 69–101
- [WK98] WEBB, Geoffrey I. ; KUZMYCZ, Mark: Evaluation of Data Aging: A Technique for Discounting Old Data During Student Modeling. In: GOETTL, Barry P. (Hrsg.) ; HALFF, Henry M. (Hrsg.) ; REDFIELD, Carol L. (Hrsg.) ; SHUTE, Valerie J. (Hrsg.): *Intelligent Tutoring Systems* Bd. 1452, 1998 (Lecture Notes in Computer Science), S. 384–393
- [WLB⁺06] WILSON, Scott ; LIBER, Oleg ; BEAUVOIR, Philip ; MILLIGAN, Colin ; JOHNSON, Mark ; SHARPLES, Paul: Personal Learning Environments: Challenging the dominant design of educational systems. In: *Proceedings of the Joint Workshop on Professional Learning, Competence Development and Knowledge Management, held in conjunction with the 1st European Conference on Technology Enhanced Learning (ECTEL 06)*, 2006, S. 67–76
- [Woe02] WOELK, Darrell: E-Learning, Semantic Web Services and Competency Ontologies. In: *World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA)*, 2002
- [Wu03] WU, Huadong: *Sensor Data Fusion for Context-Aware Computing Using Dempster-Shafer Theory*, The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, Diss., 2003
- [WW96] WILSON, T.D. ; WALSH, C.: Information behaviour: an interdisciplinary perspective. A report to the British Library Research and Innovation Centre. / British Library Research and Innovation Centre. Version: 1996. <http://informationr.net/tdw/publ/infbehav/prelims.html>. 1996. – Research report
- [WWS92] WALLS, J. ; WIDMEYER, G. ; SAWY, O. E.: Building an Information System Design Theory for Vigilant EIS. In: *Information Systems Research* 3 (1992), Nr. 1, S. 36–59
- [WZ04] WEIS, Uta ; ZINNEN, Heike: Lernen durch Wissensmanagement – Chancen und Barrieren. In: *Limpact* 8 (2004)
- [ZA05] ZELEWSKI, Stephan ; ALAN, Yilmaz: Generische Kompetenzontologie für computerbasierte Kompetenzmanagementsysteme. In: ZELEWSKI, Stephan (Hrsg.) ; ALAN, Yilmaz (Hrsg.) ; ALPARSLAN, A. (Hrsg.) ; DITTMANN, Lars (Hrsg.) ; WEICHELT, T. (Hrsg.): *Ontologiebasierte Kompetenzmanagementsysteme – Grundlagen, Konzepte, Anwendungen*. Logos, 2005

- [ZSD05] ZIEGLER, Patrick ; STURM, Christoph ; DITTRICH, Klaus R.: Unified Querying of Ontology Languages with the SIRUP Ontology Query API. In: VOSSEN, Gottfried (Hrsg.) ; LEYMAN, Frank (Hrsg.) ; LOCKEMANN, Peter (Hrsg.) ; STUCKY, Wolfried (Hrsg.): *Datenbanksysteme in Büro, Technik und Wissenschaft (BTW 2005)*, 11. GI-Fachtagung Bd. P-65. Karlsruhe, Germany, March 2-4 : Gesellschaft für Informatik, 2005 (Lecture Notes in Informatics), S. 325–344