

Ein Integrations- und Darstellungsmodell für verteilte und heterogene kontextbezogene Informationen

Dissertation

zur Erlangung des akademischen Grades

Dr. rer. pol.

vorgelegt an der

Fakultät für Wirtschaftswissenschaften
der
Technischen Universität Dresden

von

Dipl. Wirtsch.-Inf. Kevin Goslar

vorgelegt:
23. 09. 2005
verteidigt:
23. 11. 2006

Gutachter:
Prof. Dr. Eric Schoop
Prof. Dr. Alexander Schill
Prof. Dr. Rainer Thome

Inhaltsverzeichnis

1	Einordnung, Ziel und Aufbau der Arbeit	1
1.1	Einordnung	1
1.1.1	Vision: Assistenz des Menschen in der Informationsgesellschaft	1
1.1.2	Wirtschaftsinformatik	4
1.1.3	Informationsmanagement	5
1.2	Problemstellung und Ziele der Arbeit	7
1.2.1	Ausgangslage	7
1.2.2	Problemstellung	8
1.2.3	Zielstellung der Arbeit	9
1.3	Gliederung der Arbeit	10
2	Problemanalyse	11
2.1	Grundlagen	11
2.1.1	Begriffsbestimmung	11
2.1.2	Das Unternehmens- und Umweltmodell in der Betriebswirtschaftslehre	15
2.1.3	Historische Entwicklung der Kontextsensitivität	16
2.1.4	Kontextquellenarten	17
2.1.4.1	Kontextsensoren	18
2.1.4.2	Kontexterfassungssysteme	19
2.1.4.3	Kontextbewusste Anwendungen	20
2.2	Arbeitsthese	21
2.3	Verteilungsmodell für Kontextinformationen	24
2.4	Anforderungsanalyse	25
2.4.1	Bewertungsschema	25
2.4.2	Allgemeine Verwendbarkeit	25
2.4.3	Kontextbezogene Modellanforderungen	29
2.4.4	Datensicherheit	33
2.4.5	Integration heterogener Kontextinformationen	35
2.4.6	Endnutzertauglichkeit	43
2.4.7	Übersicht der Anforderungen	47
3	Betriebswirtschaftliche Potenziale umfassender Kontextsichtweisen	49
3.1	Modellseitige Effekte durch die Kombination von Kontextinformationen	49
3.2	Nutzungsformen von Kontext	51
3.2.1	Überwachung	51
3.2.2	Adaption	51
3.2.3	Assistenz	53
3.3	Kontextbezogene betriebliche Anwendungssysteme	53
3.3.1	Wearable Computing	54

3.3.2	Augmented Reality	55
3.3.3	Ubiquitous/Pervasive Computing, Ambient Intelligence	56
3.4	Anwendungsfelder von Kontextinformationen in betr. IuK-Systemen	57
3.4.1	Informationsmanagement, Information Support	58
3.4.1.1	Informationsleitstand	58
3.4.1.2	Kontextbezogenes Informationsmanagement	59
3.4.1.3	Information Retrieval	59
3.4.1.4	Persönliches Informationsmanagement	61
3.4.1.5	Kontextsensitive Verzeichnisse	62
3.4.1.6	Kontextsensitive Informationsdarstellung	62
3.4.1.7	Kontextbezogene Informationsorganisation	63
3.4.2	Entscheidungsunterstützung	63
3.4.3	Wissensmanagement	64
3.4.3.1	Organisationale Perspektive	65
3.4.3.2	Technische Perspektive	65
3.4.3.3	Organisational Memory Systeme	65
3.4.3.4	Erfahrungsmanagement	66
3.4.3.5	Fazit	67
3.4.4	Dokumentation/Logging	67
3.4.5	E-Learning, Computer aided Instruction	68
3.4.6	Administration von Anwendungssystemen	69
3.4.7	Fehlertolerante Systeme und Prozesse	70
3.4.8	Lagerhaltung	72
3.4.9	Gebäudemanagement	73
3.4.10	Prozessmanagement, Workflow Computing	74
3.4.11	Teamwork-Support, Workgroup Computing	75
3.4.12	Mobile Informations- und Kommunikationsunterstützung	76
3.4.12.1	Kontextsensitive Telekommunikation	76
3.4.12.2	Kontextsensitive Face-to-Face-Kommunikation	77
3.5	Technologische Effekte automatisiert verfügbarer Kontextinformationen	78
3.5.1	Erhöhung der Informationsbasis über Realweltobjekte	78
3.5.2	Teilautomatisierung der Erkennung und Handhabung von Problemen	78
3.5.3	Effizientere Gestaltung der Mensch-Maschine-Schnittstelle	78
3.5.3.1	Verschlinkung	79
3.5.3.2	Adaption	80
3.5.4	Erhöhung des Automatisierungsgrades	81
3.5.4.1	Reaktivität	83
3.5.4.2	Proaktivität	84
3.5.5	Unterstützung von Echtzeitverhalten	84

3.6	Wirtschaftliche Effekte einer automatisierten Versorgung betrieblicher Aufgabenträger mit Kontextinformationen	85
3.6.1	Engere Kundenorientierung	85
3.6.1.1	Mass Customization	85
3.6.1.2	C-Marketing	86
3.6.1.3	C-Marktforschung	87
3.6.1.4	Kontextunterstützter Point-of-Sale	87
3.6.2	Flexibleres Controlling	88
3.6.3	Effizientere Wartung	88
3.6.4	Effektivere Nutzung der betrieblichen Wissensbasis	89
3.6.5	Leistungsbezogenere Vergütung	89
3.6.6	Erhöhung der Transparenz	91
3.6.7	Effektivere Problemidentifikation	91
3.6.8	Verringerung von Prozessdurchlaufzeiten	91
3.7	Kontextbezogene Dienstleistungsformen	92
3.7.1	Allgemeine Verwendungsmöglichkeiten	93
3.7.2	Verwendung von Kontext in der Anbahnungsphase	94
3.7.3	Verwendung von Kontext in der Vereinbarungsphase	95
3.7.4	Verwendung von Kontext in der Abwicklungsphase	95
3.8	Zusammenfassung	95
4	Verwandte Arbeiten	97
4.1	Formen von Kontextvariablen	97
4.1.1	Stellvertreteransatz	97
4.1.2	Kontextebenenmodelle	98
4.1.3	Räumliche Weltmodelle	99
4.1.3.1	Symbolische Ortsmodelle	100
4.1.3.2	Geometrische Ortsmodelle	100
4.1.4	Semantische Weltmodelle	101
4.1.5	Situationsmodelle	101
4.1.5.1	Context Schemas	101
4.1.5.2	Information Logistics	102
4.1.5.3	TEA	102
4.2	Integration heterogener Informationsmengen	103
4.2.1	Unternehmensweite Datenmodelle	103
4.2.2	Enterprise Application Integration	104
4.2.3	Operational Data Store	105
4.2.4	Semantic Web	106
4.3	Integration aktiver Elemente in Datenstrukturen	108
4.3.1	Externe Agenten	108
4.3.2	Objektorientierung	109

4.3.3	Aktive Assoziationen	110
4.3.4	Transformatorenpfade	110
4.4	Historisierung von Informationen	111
4.5	Informationsmetamodelle	112
4.5.1	Relationales Datenmodell	113
4.5.2	Objektorientiertes Datenmodell	114
4.5.3	XML	114
4.5.4	RDF	115
4.5.5	Topic Maps	116
4.5.6	Ontologien	117
4.6	Kontextmodelle	118
4.6.1	Context Cubes	118
4.6.2	Henricksen	119
4.6.3	Context Toolkit	121
4.6.4	Context Fabric	122
4.6.5	Context Shadow	124
4.6.6	Sentient Computing	125
4.6.7	Context Framework	127
4.6.8	Gaia	128
4.6.9	Tupelraumbasierte Ansätze	130
4.6.10	Auswertung	131
4.7	Zusammenfassung	134
5	Context-Maps	135
5.1	Informationsintegrationsmodell	135
5.1.1	Zugriffsmodell	135
5.1.2	Integrationsmodell	136
5.1.3	Verbindungsmodell	137
5.1.4	Datenverteilungsaspekte	137
5.1.5	Gesamtarchitektur	139
5.2	Kontextmodell	140
5.2.1	Grundlegende Kontextmodellelemente	140
5.2.2	Universen	140
5.2.3	Ontologien	142
5.2.4	Szenen	142
5.2.5	Situationen	143
5.3	Wiederverwendungsmodell	144
5.3.1	Persistente Realweltobjektmodelle	144
5.3.2	Invariante Situationsaspekte	144
5.4	Elemente von Context Maps	144
5.4.1	Basis: Topic Maps	145

5.4.2	Topics	146
5.4.2.1	Namensgebung	147
5.4.2.2	Identifikation	148
5.4.3	Occurrences	149
5.4.4	Attribute	151
5.4.4.1	Beschaffungsmap	152
5.4.4.2	Historie	154
5.4.4.3	Qualitätsdeskriptoren	155
5.4.4.4	Sicherheitsdeskriptoren	155
5.4.4.5	Dynamikdeskriptoren	157
5.4.5	Assoziationen	157
5.4.5.1	Normale Assoziationen	158
5.4.5.2	Abstrakte Assoziationen	158
5.4.5.3	Aktive Assoziationen	161
5.4.6	Scope	161
5.5	Schichtenarchitektur	162
5.5.1	Ontologieebene	163
5.5.2	Szenenebene	165
5.5.3	Realweltmodellebene	167
5.5.4	Externe Datenebene	168
5.5.5	Kontextquellentypenebene	169
5.6	Richtlinien zur Benutzung	170
5.6.1	Strukturelle Veränderung von Universen	170
5.6.2	Transformation in andere Ontologien	171
5.6.3	Abfrage von Kontextinformationen	172
5.7	Anwendungsformen des Modells	172
5.7.1	Informationsbasis über Realweltobjekte	172
5.7.2	Situationsmodellierung	173
5.7.3	Interaktionsplattform für kontextsensitive Entitäten	173
5.8	Zusammenfassung	174
6	Validierung	175
6.1	Realisierbarkeit	175
6.2	Funktionale Validierung	175
6.2.1	Nutzerprofil	176
6.2.2	Messenger-VUI	179
6.2.3	Restaurant-Finder	181
6.2.4	Kontextsensitives Navigationssystem	186
6.2.5	Tourist Guide	193
6.3	Konzeptionelle Validierung	195
6.3.1	Wissenschaftliches Ergebnis	195

6.3.2	Anforderungsanalyse	196
6.4	Problemdiskussion	197
6.4.1	Unmöglichkeit der Nutzenquantifizierung	197
6.4.2	Schutz der Privatsphäre	199
6.4.3	Zusätzlicher Modellierungsaufwand	202
6.4.4	Integration in existierende Umgebungen	202
6.4.5	Gesellschaftlicher Paradigmenwechsel	203
6.4.6	Überformalisierung	204
6.4.7	Skalierbarkeit des Modells	204
6.4.8	Performance	205
6.5	Zusammenfassung	205
7	Zusammenfassung und Ausblick	207
7.1	Zusammenfassung	207
7.2	Ausblick	208
	Abbildungsverzeichnis	I
	Tabellenverzeichnis	V
	Abkürzungsverzeichnis	VII
	Literaturverzeichnis	IX
	Anhänge	XXXVII
A	Detaillierte Bewertungen verwandter Arbeiten	XXXVII
A.1	Entity-Relationship-Modell	XXXVII
A.2	Objektorientiertes Datenmodell	XL
A.3	XML	XLII
A.4	Semantic Web (RDF)	XLIV
A.5	Topic Maps	XLVI
A.6	Context Cubes	XLVIII
A.7	Henricksen	L
A.8	Context Toolkit	LII
A.9	Context Fabric	LIV
A.10	Sentient Computing	LVI
A.11	Context Framework	LVIII
A.12	Gaia	LX
A.13	Schmidt	LXII
A.14	Context Maps	LXIV

B XML-Notation für Context Maps	LXVII
B.1 Context Map	LXVII
B.2 Realweltuniversum	LXVII
B.3 Instanzen	LXVII
B.4 Kontextquellenebene	LXVIII
B.5 Allgemeine Context-Map-Elemente	LXVIII
B.5.1 Maps	LXVIII
B.5.2 Topic	LXVIII
B.5.3 Attribute	LXVIII
B.5.4 Normale Assoziationen	LXIX
B.5.5 Abstrakte Assoziationen	LXX
B.5.6 Aktive Assoziationen	LXX
C URI-Notation für Occurrences	LXXI
D Regeln zur Wohlgeformtheit von Context Maps	LXXIII
E Anwendungskontextmodelle	LXXIV
E.1 Generische Grundlage	LXXIV
E.1.1 Ontologien	LXXIV
E.1.2 Szenen	LXXV
E.2 Nutzerprofil	LXXVII
E.2.1 Ontologien	LXXVII
E.2.2 Szenen	LXXIX
E.2.3 Kontextquellentypenebene	LXXXV
E.2.4 Realweltmodellinstanz	LXXXVI
Ehrenwörtliche Erklärung	XC

1 Einordnung, Ziel und Aufbau der Arbeit

Wahrnehmung und Erfassung von Situationen sowie das Handeln mit Kontextbezug ist für den Menschen ein wesentlicher Bestandteil im alltäglichen Umgang mit Information und Kommunikation.

Albrecht Schmidt
Lancaster University

1.1 Einordnung

Information und Wissen sind eine wichtige Grundlage der heutigen Gesellschaft geworden und spielen eine zentrale Rolle in all ihren Bereichen, d. h. im öffentlichen und privaten Leben, Wirtschaft und Verwaltung, weswegen oft von einer *Informationsgesellschaft* die Rede ist (vgl. [Haef80, Bang95, Dieb96, Schu01]). Informationsbezogene Tätigkeiten dominieren mehr als 80 Prozent der heutigen Arbeitsplätze, für die nahe und mittlere Zukunft werden weitere drastische Steigerungen prognostiziert (vgl. u. a. [BMWi95, BMWi03]). Nahezu alle gesellschaftlichen Bereiche werden durch jederzeit und überall verfügbare informationsverarbeitende Technologien, die unter dem Schlagwort *Evernet* zusammengefasst werden, durchdrungen und umgestaltet (vgl. [Bers00]). Ziel der Forschung in diesem Bereich ist es, nicht immer weitere kompliziertere Technologien zu entwickeln und damit die Gesellschaft gegen ihren Willen zu *informatisieren*, sondern durch menschenfreundlich gestaltete Technologien die informationsbezogenen Mehrwerte für eine *informierte* Gesellschaft nutzbar zu machen, d. h. den Umgang mit Informationen intuitiv, einfach und transparent zu gestalten und den Menschen in seiner freien Entfaltung in dieser Informationsgesellschaft zu assistieren (vgl. [Haef80, Bühl97]).

1.1.1 Vision: Assistenz des Menschen in der Informationsgesellschaft

Die Tätigkeiten des Menschen vollziehen sich verteilt innerhalb der menschlichen Lebenswelt (im folgenden *Realwelt* genannt, genaueres in Kap. 2.1.1 („Begriffsbestimmung“, S. 11)), und mit Bezug auf die Realwelt, weshalb die Realwelt einen wesentlichen Einfluß auf die Tätigkeiten des Menschen ausübt, d. h. sie determiniert zu einem wesentlichen Teil die Handlungen des Nutzers (vgl. Relation (1) und (2) in Abb. 1 auf der nächsten Seite). Die verschiedenen Aspekte der Realwelt, die in verschiedenen Situationen einen unterschiedlich relevanten Einfluß auf die Tätigkeit des Menschen ausüben, werden dabei vom menschlichen Nutzer automatisch berücksichtigt, d. h. der Mensch ist „kontextsensitiv“, denn dieser Kontextbezug ist für viele von Menschen bearbeitete Aufgabenstellungen notwendig. Da der Mensch seine Tätigkeiten in der Realwelt mittels Werkzeugen vereinfacht, müssen diese Werkzeuge folgerichtig ebenfalls ein situationsabhängiges Verhalten aufweisen (vgl. Relation (3) und (4) in Abb. 1 auf der nächsten Seite). Dieses Prinzip des Kontextbezuges von Werkzeugen wird als *Kontextsensitivität* bezeichnet.

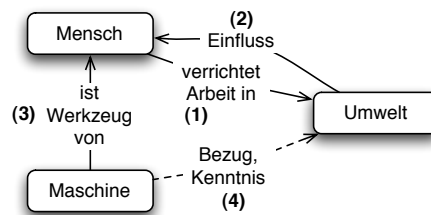


Abbildung 1: Umweltbezug von menschlichen Anwendern und deren Werkzeugen

Computersysteme haben sich als sehr geeignete und universelle mentale Werkzeuge erwiesen, weswegen sie menschlichen Nutzern in immer weiteren Bereichen des menschlichen Tätigkeitsspektrums assistieren (vgl. [KrSi99, S. 368]). Um dem Menschen bei möglichst vielen Tätigkeiten in seiner Lebenswelt zur Hand gehen zu können, sind deshalb Erweiterungen der Fähigkeiten von Computertechnik analog – aber nicht identisch – zu den Fähigkeiten von Menschen nötig (vgl. Abb. 2). Dazu zählen neben der Fähigkeit, eigenständig Informationen aufzunehmen (Kontextsensitivität) und dazu Lösungsvorschläge zu finden (maschinelles Lernen und Künstliche Intelligenz) auch, diese ggf. nach Absprache mit dem menschlichen Benutzer eigenständig umzusetzen (Adaption, Robotik). Weiterhin werden Mechanismen benötigt, um eigene Erfahrungen zu speichern und bei passender Gelegenheit zu erinnern (Daten- und Wissensbanken) sowie zum Zweck der Koordination und Überwindung der subjektiven Sichtweise – dazu zählen z. B. die Vervollständigung und Fehlerbereinigung des eigenen Erfahrungsschatzes durch Vergleich der eigenen Erfahrungen mit denen anderer Subjekte – mit anderen Systemen auszutauschen (Kommunikation durch Rechnernetze).

Hinreichend *intelligente* Assistenten, die auch in unstrukturierten, innovativen Situationen eine sinnvolle Assistenz ähnlich einem aktiv mitdenkenden menschlichen Assistenten erbringen können, werden erst entstehen, wenn sämtliche in Abb. 2 dargestellten Erweiterungen hinreichend entwickelt und miteinander kombiniert werden (vgl. [McC⁺55, S. 12]). Es entsteht ein Szenario, in dem Verfahren des maschinellen Lernens durch Verfahren der Kontexterfassung und -darstellung mit einer derart umfang-

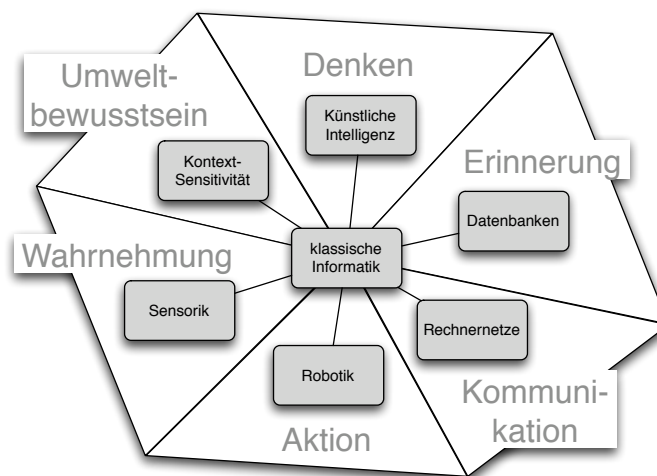


Abbildung 2: Funktionale Erweiterungen von Computersystemen

reichen Menge von Realweltinformationen versorgt werden, dass sie wirklich eigenständig lernen, d. h. neue Thesen entwickeln und dank Robotik auch selbst verifizieren können. Drahtlose, ständig verfügbare Kommunikationswege und leistungsfähige Daten- und Wissensbanken stellen in diesem Szenario den Austausch und damit die Kollektivierung des Wissens aller Assistenzsysteme sicher, so dass der Erfahrungsschatz von maschinellen Assistenten durch Kombination der Einzelerfahrungen verschiedener Anwendungen und Assistenten analog zur menschlichen Wissenschaft schnell einen enormen Umfang erreicht. Solange Computer jedoch in ihrer heutigen Form als Kasten unter dem Schreibtisch stehen und die Welt nur über die spärlichen, von ihrem jeweiligen Nutzer mittels Tastatur und Maus bereitgestellten Informationen erleben, werden keine grundlegenden Fortschritte auf dem Gebiet des maschinellen Lernens und der intelligenten Assistenz des Menschen möglich werden, um dem Computer das für selbstständige Tätigkeiten notwendige Weltwissen zu vermitteln. Selbiges gilt ja auch für den weitaus lernfähigeren Menschen. Wenn man ihn in eine Kiste unter dem Schreibtisch eines Anwenders einsperren und nur per Tastatur mit seiner Umwelt kommunizieren lassen würde, wäre seiner Intelligenz und Nützlichkeit in praktischen Situationen eine ähnliche Grenze gesetzt wie heutigen Computersystemen. Ein Beispiel stellen die seit einem Jahrzehnt andauernden, aber nur sehr bedingt erfolgreichen Arbeiten an CYC dar (vgl. u. a. [Lena95]).

Eine der wichtigsten Ursachen für diese Einschränkung der Leistungsfähigkeit von heutigen Computeranwendungen ist der Engpaß hinsichtlich der Aufmerksamkeit, die ein Nutzer bereit ist, einem Computersystem zu widmen (vgl. [Ebl⁺01, S. 1]). Zwar könnten nahezu alle Aufgaben durch einen geduldigen Nutzer interaktiv am Computer realisiert werden, doch sind in der Praxis nur die wenigsten Nutzer bereit bzw. haben Zeit, sämtliche in diesem Zusammenhang benötigten Informationen manuell einzugeben. Fortgeschrittene Anwendungsszenarien scheitern oft an dem damit verbundenen zusätzlichen und komplexeren Bedienungsaufwand durch den Nutzer. Daher beschränkt sich die Funktionalität klassischer maschineller Assistenzsysteme, die lediglich die vom menschlichen Nutzer vorgegebenen Informationen und Anweisungen verarbeiten, auf die möglichst schnelle und automatisierte Unterstützung standardisierter Teilaufgaben, etwa die Umwandlung von gegebenen Informationen in andere Repräsentationen oder die Übertragung und Speicherung großer Mengen von Informationen. Untermauert wird diese Argumentation durch den ausgebliebenen Durchbruch von Wissensmanagementsystemen, welche eine zwar wichtige Funktion adressieren, sich aber in der Praxis nie richtig durchsetzen konnten, da die Wissensträger im betrieblichen Alltag keine Zeit haben, ihr Wissen explizit in einen Computer einzugeben, besonders dann nicht, wenn sie dieses Wissen unter Zeitdruck generieren. Die einzig mögliche Lösung zur Steigerung der Funktionalität von computergestützten Anwendungssystemen, ohne gleichzeitig die Anforderungen an die dafür benötigte menschliche Aufmerksamkeit zu erhöhen, stellt der Ansatz dar, durch explizite Sensoren die Anzahl der Eingabekanäle in Computersysteme und damit die Menge an *impliziten* Eingabedaten drastisch zu erhöhen, d. h. Computersysteme in die Lage zu versetzen, sich selbsttätig die für ihre Funktionalität benötigten Informationen zu besorgen. Kontextsensitivität soll also den Informationsmangel in Computersystemen beheben, der entsteht, weil Computer im klassischen Szenario auf die Informationen angewiesen sind, die der menschliche Benutzer ihnen *explizit* mitteilt. Was der Mensch dem Computer nicht mitteilt – was seine Ursache in Trägheit, Unwissenheit,

Vergesslichkeit oder gar Böswilligkeit sowie in den quantitativen¹ und qualitativen² Eigenschaften der Datenbasis begründet sein kann – ist für den Computer nicht existent. Dieser Informationsmangel und die daraus entstehende zu grobe Weltsicht kann die Ursache für fatale Fehlentscheidungen seitens der automatisierten betrieblichen Informationssysteme darstellen. Einen weiteren, bereits angedeuteten Entwicklungsschritt stellt die Integration bisher abgegrenzter Informationsmengen aus einzelnen Dimensionen des unterstützenden menschlichen Diskursbereiches dar, was zu einem umfassenderen Überblick über die vom menschlichen Benutzer tatsächlich bearbeiteten Realweltvorgänge und -probleme führen soll.

Diese Arbeit stellt einen Ansatz vor, um die durch verschiedene Anwendungen ermittelten Kontextinformationen in ein umfassendes Bild über die Umwelt, praktisch eine „semantische Landkarte“ des Kontextes, zu integrieren, um sie allen interessierten Anwendungen zur Verfügung zu stellen und ist damit in den Bereich der systematischen Informationsbereitstellung für menschliche und vor allem maschinelle Aufgabenträger einzuordnen. Dabei wird diese Domäne schwerpunktlässig aus der Sichtweise der *Wirtschaftsinformatik*, insbesondere des *Informationsmanagements* betrachtet, welche aufgrund des fächerübergreifenden Charakters dieser Arbeit im folgenden kurz skizziert werden.

1.1.2 Wirtschaftsinformatik

Obwohl ein einheitliches Grundverständnis des Faches bzw. der Aufgaben, Forschungsfragen und Erkenntnisobjekte der Wirtschaftsinformatik zur Zeit nur bedingt vorhanden sind, lassen sich grundlegende Leitbilder dieser Wissenschaft identifizieren (vgl. [Lehn99, S. 7]). Neben der Erklärung und Gestaltung der Informationsfunktion in betrieblichen und gesellschaftlichen Domänen gehört dazu die Unterstützung bei der Erfüllung betrieblicher Aufgabenstellungen durch die Bereitstellung von betrieblichen (d. h. in Wirtschaft oder Verwaltung eingesetzten) Informations- und Kommunikationsstrategien und -systemen, welche auch IT-Systeme oder betriebliche Anwendungssysteme genannt werden und als sozio-technische Systeme sowohl menschliche als auch maschinelle Aufgabenträger umfassen (vgl. [FeSi98, S. 65 f]). Hintergrund ist die Unterstützung von Planungs-, Steuerungs- und Kontrollaufgaben sowie der dazugehörigen Kommunikation innerhalb bzw. mit der Umwelt der Unternehmung (vgl. [Scho04, S. 718 ff]). Neben der durch ein wissenschaftliches Theoriengebäude ermöglichten bzw. unterstützen Analyse und Beschreibung der in diesem Umfeld auftretenden Situationen und Probleme geht es dabei um die gleichermaßen theoretisch fundierte Konzeption von entsprechenden Lösungen, welche typischerweise eine Art von maschineller Informationsverarbeitung beinhalten, was sich auch in der Namensgebung dieser Wissenschaft widerspiegelt, sowie deren praktische Umsetzung, wobei weniger die *Entwicklung* von Basistechnologien, sondern vielmehr die *Erschließung* der dadurch entstehenden Potenziale und Veränderungen im betrieblichen Umfeld im Vordergrund steht (vgl. u. a. [Groc75, Preß80, SzEi83, Sche88, Hein93, Mer⁺01]).

Das mit den Forschungszielen der Wirtschaftsinformatik kongruente Forschungsziel dieser Arbeit ist neben der Bewertung und Einordnung technischer Konzepte in betriebswirtschaftliche Zusammenhänge die Entwicklung eines generischen Modells, um Anwendern die strukturierte Problemlösung im Bereich

¹es müssen zu viele Daten erfasst werden

²die Genauigkeit und Aktualität der zu erfassenden Daten übersteigt die menschlichen Fähigkeiten

der Beschaffung von Kontextinformationen zu ermöglichen. Dieses Modell geht über technische Aspekte hinaus und beinhaltet betriebswirtschaftliche Anwendungsaspekte.

1.1.3 Informationsmanagement

Information und Wissen als Teile des Produktionsfaktors *dispositive Arbeit* sind in der modernen Wirtschaft sowohl qualitativ als auch quantitativ derartig bedeutend, dass sie neben den klassischen drei Produktionsfaktoren (Arbeit, Boden und Kapital, vgl. [Smit75, Gute51]) oft als vierter dispositiver Produktionsfaktor bezeichnet werden, der im Gegensatz zu den anderen Produktionsfaktoren heute in noch nicht ausreichendem Maß erschlossen ist und deshalb im Brennpunkt vielfältiger wissenschaftlicher Untersuchungen steht (vgl. [Öste87, ZaRü89, HaSc94, Kön⁺95, Bode97]). Information stellt eine notwendige Grundlage für die Koordination arbeitsteiliger Leistungsprozesse sowie zum Treffen fundierter Entscheidungen im betrieblichen Produktions- und Leitungsprozess dar und wird daher in unterschiedlicher Form zu jeder Zeit an allen Stellen im Betrieb benötigt (vgl. [Mer⁺01, S. 4 f]). Dabei steigt der Koordinationsaufwand mit zunehmender Arbeitsteilung und Spezialisierung, d. h. der Ausrichtung einzelner Unternehmen, Branchen und Märkte auf immer speziellere Abschnitte der Wertschöpfungskette, drastisch an. Transaktionskosten, d. h. die bei der zwischenbetrieblichen Koordination entstehenden Kosten für Information und Kommunikation (IuK), Entscheidung, Verhandlung und Kontrolle sind zwischen 1870 und 1970 von 25 % auf 55 % des Bruttosozialproduktes einer Industrienation gestiegen und dürften heute mehr als 60 % umfassen (vgl. [WaNo86, S. 121], [Coas37]). Dies verdeutlicht die Relevanz des Produktionsfaktors *Information* sowohl auf der einzel- als auch der gesamtwirtschaftlichen Ebene einer modernen Informationsgesellschaft. Neben neuartigen, flexiblen betrieblichen Organisationsformen sind dabei vor allem effektivere Informations- und Kommunikationssysteme (IuK-Systeme) und -verfahren nötig, um die durch stärkere Spezialisierung entstehenden Produktivitätsgewinne nicht durch weiter steigende Informations- und Kommunikationskosten zu kompensieren. Information stellt also einen virtuellen, aber nichtsdestotrotz wesentlichen Produktionsfaktor innerhalb des Unternehmens dar, mit dem die Aufgabenträger im betrieblichen Umfeld individuell und bedarfsgerecht, also in passender Menge, Form und Qualität, zur richtigen Zeit, am richtigen Ort zu versorgen sind (vgl. [Ott92, S. 476 ff], [PiFr93, S. 433], [HaSc94], [Sonn99, S. 96]).

Während hauptsächlich personell realisierte Aufgabenbereiche durch betriebswirtschaftliche Methodiken adressiert werden, ist die Organisation der Informationsverarbeitung durch einen personell lediglich gestützten maschinellen Aufgabenträger in die betriebliche Querfunktion des *Informationsmanagements* einzuordnen (vgl. [Scho04, S. 720], [Nast89, Hein99, Krcm97]). Das *Informationsmanagement* als Teilbereich der Wirtschaftsinformatik stellt ein auf wissenschaftliche Methoden aufbauendes Leitungshandeln dar, welches als Sachziel das Leistungspotenzial der Informationsfunktion in Unternehmen durch die Schaffung einer geeigneten *Informationsstrategie*, die den Umgang mit Informationen sowohl auf der betriebswirtschaftlichen, d. h. der organisatorischen und personellen, als auch auf der technischen Ebene regelt und die Schaffung einer sozio-technischen Informationsinfrastruktur sowie Informationskultur beinhaltet, voll entfalten und letztlich in Unternehmenserfolg umsetzen soll (vgl. [ZaRü89, S. 34 ff], [Hein99, S. 21]). Durch die letztgenannten Maßnahmen sollen die betrieblichen Aufgabenträger indivi-

duell und bedarfsgerecht mit den notwendigen Informationen versorgt werden, um die Qualität der von diesen Aufgabenträgern zu treffenden Entscheidungen sicherzustellen (vgl. u. a. [Ott92]).

Der hierbei deutlich werdende individuelle, situationsbezogene Aspekt der Informationsversorgung macht das Informationsmanagement an sich zu einer kontextbezogenen Disziplin, die auf Kontextinformationen angewiesen ist, um beliebige andere Informationen in der bereits dargestellten kontextbezogenen Weise zustellen zu können, denn Kontextinformationen bilden ergänzend zur Betriebsdatenerfassung, die sich an internen Produktionsprozessen und deren Parametern orientiert, äußere Realweltaspekte von betrieblichen Prozessen ab und können somit als Grundlage für eine ganze Reihe betriebswirtschaftlicher Entscheidungen und Konzepte dienen. Auf diesen Aspekt wird in Kap. 3 detaillierter eingegangen. Damit Kontextinformationen zur Lenkung anderer Informationsflüsse bereitstehen, müssen sie selbst durch ein funktionierendes *Kontextinformationsmanagement* in angemessener Form bereitgestellt werden. Kontextinformationen sind in dieser Hinsicht also sowohl als Steuerungskomponente als auch als gesteuerter Inhalt von Informationsmanagementprozessen aufzufassen. Damit stellt die Verarbeitung und Integration von Kontextinformationen ein Realweltproblem aus der Domäne des Informationsmanagements dar. Durch Schaffung eines geeigneten sozio-technischen *Kontextinformationssystems* soll die Beschaffung, Strukturierung und Aufbereitung intern und extern gewonnener Kontextinformationen mit dem Ziel, die richtigen Kontextinformationen im richtigen Umfang, in der richtigen Form, zur richtigen Zeit, in der richtigen Qualität als Unterstützung einer automatisierten und nicht-automatisierten Informationsverarbeitung für Transaktions- und Entscheidungsaufgaben bereitgestellt werden (vgl. [Stoc00]). Konkret müssen Kontextinformationen aus existierenden Informationsbeständen beschafft und entsprechend den jeweiligen betrieblichen Aufgabenbereichen und -strukturen organisiert und verteilt werden. Im Gegensatz zu anderen Informationsformen sind Kontextinformationen sehr schnell veränderlich und veralten sehr schnell, weshalb sie mehr als andere Informationsarten in Unternehmen durch technische Aufgabenträger verwaltet werden müssen. Ein solches noch zu erarbeitendes technisches Kontextinformationssystem stellt also einen notwendigen Teil eines umfassenden Konzeptes zum betrieblichen und technischen *Kontextinformationsmanagement* zum Abbau von Informationsbarrieren in diesem Bereich dar (vgl. [Holl99]). Menschliche Aufgabenträger werden in diesem Zusammenhang vor allem zur Steuerung, aber nicht Durchführung dieser Vorgänge benötigt.

Neben dem eigentlichen Inhalt stellt also die *Struktur* der bereitgestellten Informationen eine wesentliche Voraussetzung für ein erfolgreiches Kontextinformationsmanagement dar, da die dargestellte Informationsverteilung nur mittels geeignet strukturierter Informationsbestände erfolgen kann (vgl. [RaSv95, S. 75 ff], [Bode97, S. 461 ff], [MeZa96, S. 46 ff]). HARVEL verdeutlicht diesen Sachverhalt mit der Aussage „*as the amount of stored context grows, we are reaching the point where we are data rich, but information poor*“ [Har⁺03, S. 3]. Wie in dieser Arbeit noch aufgezeigt wird, besteht in Unternehmen ein potentieller Bedarf nach integrierten heterogenen Informationen. Wie ebenfalls dargestellt wird, fehlt ein geeignetes Informationsmodell, um aus heterogenen Datenquellen stammende kontextbezogene Informationen in einer integrierten Form darzustellen, womit ein effektives und effizientes Kontextinformationsmanagement unmöglich wird. Daher wird in dieser Arbeit ein Integrations- und Darstellungsmodell für Kontextinformationen als Teil eines Paradigmas zur systematischen Beschaffung, Auswahl, Integration, Organisation und Verteilung von Kontextinformationen durch sozio-technische Kontextin-

formationssysteme im Rahmen einer unternehmensweiten Kontextinformationsstrategie entwickelt. Die dadurch realisierte Kontextinformationsversorgung stellt eine Dienstleistung innerhalb des Unternehmens zur Unterstützung betriebswirtschaftlicher Planungs-, Entscheidungs- und Steuerungsvorgänge dar.

1.2 Problemstellung und Ziele der Arbeit

In dieser Arbeit wird ein Verfahren zur Integration der verteilten und heterogenen kontextbezogenen Informationsbasis eines Unternehmens entwickelt. Zunächst wird dargestellt, dass Kontextinformationen im betrieblichen Umfeld prinzipbedingt an vielen Stellen verteilt entstehen und ebenso verteilt genutzt werden. Aufbauend auf dieser Erkenntnis wird ein Verfahren zur Sammlung, Integration und Verteilung von Kontextinformationen von und zu verschiedenen Teilsystemen im betrieblichen Anwendungssystem entwickelt. Typische Einsatzgebiete der entwickelten Technologie werden aufgezeigt, funktionale sowie ökonomische Potenziale in betrieblichen Anwendungssystemen skizziert.

1.2.1 Ausgangslage

Damit die meisten kontextbezogenen Assistenzsysteme im praktischen Einsatz wirklich sinnvolle Assistenzleistungen erbringen können, sind umfangreiche Kontextinformationen verschiedener Art notwendig, die über die heute verwendeten elementaren Kontextdimensionen wie den Ort der beteiligten Objekte hinausgehen. Neben detaillierten Informationen über den jeweiligen Benutzer gehören dazu nicht nur Informationen über die Identität anderer beteiligter Kontextobjekte, sondern auch Informationen über den Kontext dieser Kontextobjekte sowie die aus der Kombination verschiedener Kontextinformationen resultierenden Schlußfolgerungen. Ein Beispiel zeigt Beispiel 1.

Beispiel 1 *Damit eine Restaurant-Finder-Anwendung, welche geeignete Restaurants in der Nähe des Nutzers sucht, wirklich brauchbare Ergebnisse und nicht nur eine u. U. sehr lange Liste von Restaurants in der Nähe darstellt, sind neben dem Ort des Benutzers und den Namen der betreffenden Restaurants Informationen über deren Öffnungszeiten, gegenwärtige Warteschlange³, Menü, Tagesspezialität sowie möglicherweise zu beachtende Präferenzen aller beteiligten Personen, etwa hinsichtlich der Rollstuhltauglichkeit des Restaurants, zu beachten.*

Wie aus Beispiel 1 ersichtlich wird, stammen die in fortgeschrittenen kontextsensitiven Anwendungen verwendeten Kontextinformationen aus einer Vielzahl von Datenquellen, etwa Lokalisierungssystemen, welche den Nutzer oder eines seiner mobilen Endgeräte lokalisieren, den aus dem Ort abgeleiteten Gelben Seiten der betreffenden Stadt sowie den per Internet zugreifbaren Geschäftsanwendungen der beteiligten Unternehmen. Es ergibt sich ein Szenario aus diversen Quellen von Kontextinformationen, die von kontextsensitiven Anwendungen zum Zweck der Beschaffung von Kontextinformationen zugegriffen werden müssen.

Herkömmliche kontextsensitive Anwendungssysteme realisieren die Aspekte des Zugriffs und der Darstellung der von ihnen benötigten Kontextinformationen selbst. Diese Herangehensweise wird im eben dargestellten Szenario mit vielen Datenquellen und kontextsensitiven Anwendungen zum Problem,

³Zu volle oder leere Restaurants sollen vermieden werden.

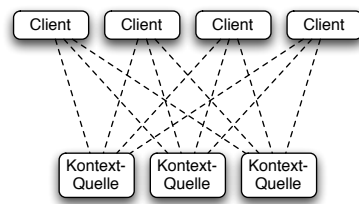


Abbildung 3: Das Verteilungsproblem bei heterogenen Kontextquellen

da die verschiedenen Zugriffs- und Transformationsverfahren dann redundant in jeder Anwendung implementiert werden müssen. Benötigt wird ein Verfahren, welches die verschiedenen Kontextinformationsverbraucher in die Lage versetzt, die von ihnen benötigten Kontextinformationen sowie ggf. deren Datenquellen zu definieren und diese Informationen daraufhin aus einem *digitalen Abbild* der Realwelt zu entnehmen, in dem sämtliche bekannten Informationen über die Realwelt, welche aus beliebigen Datenquellen stammen können, verzeichnet sind (vgl. [Schm02, S. 88]). Eine detailliertere, formale Analyse der Problemdomäne findet sich in Kap. 2.3 („Verteilungsmodell für Kontextinformationen“, S. 24).

1.2.2 Problemstellung

Das in der vorliegenden Arbeit thematisierte Problem ist die Überwindung der *Informationslücke*, die zwischen dem Bedarf an Kontextinformationen auf Seiten der kontextsensitiven Anwendungen einerseits und den von den verschiedenen Kontexterfassungssystemen bereitgestellten Kontextinformationen auf der anderen Seite besteht. Beide Informationsmengen unterscheiden sich hinsichtlich ihrer Strukturierung, Nomenklatur, Mächtigkeit und Granularität. Kontextsensitive Anwendungen sind an Informationen über Realweltobjekte interessiert und müssen sich diese bisher selbsttätig aus den einzelnen Kontextquellen zusammensuchen und integrieren, was einen fehlerträchtigen und redundanten Informationsbeschaffungsprozess darstellt (vgl. [Hong01, S. 3]). Insbesondere ergeben sich die folgenden beiden Probleme.

Problem 1 (Verteilungsproblem): *kontextsensitive Anwendungen müssen mehrere Kontextquellen nach Kontextinformationen befragen.*

Dazu müssen sie nicht nur über die Existenz jeder einzelnen Kontextquelle Bescheid wissen, sondern diese auch zugreifen und die dort verwendeten proprietären Zugriffsprotokolle und Ontologien interpretieren können. Dies kann in komplexen Anwendungsszenarien mit voneinander durch Sicherheitsmaßnahmen abgeschotteten Infrastrukturen sowie dutzenden Kontextquellen und Hunderten Clients nicht sichergestellt werden und stellt ein technisch wie organisatorisch nichttriviales Problem dar (vgl. Abb. 3).

Problem 2 (Integrationsproblem): *die Zusammenhänge zwischen den in verschiedenen Kontextquellen befindlichen Kontextinformationen sind nicht modelliert.*

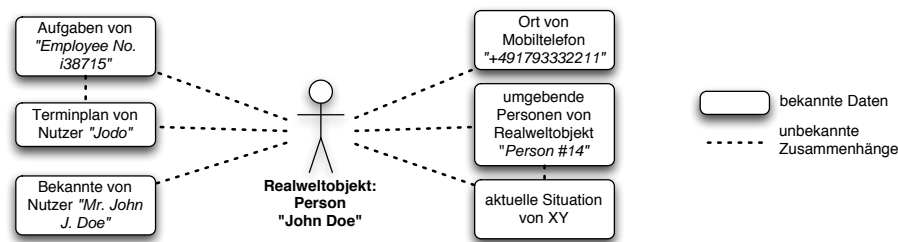


Abbildung 4: Darstellung des Integrationsproblems

Da die einzelnen Kontextdimensionen in der Praxis prinzipbedingt von verschiedenen Kontextquellen bereitgestellt werden und jede Kontextquelle ihr eigenes Kontextmodell verwendet, gibt es mehrere separate, allesamt unvollständige Kontextmodelle, die jedoch alle die selbe Realwelt beschreiben. Beispielsweise stellt ein Lokalisierungssystem den augenblicklichen Ort einer Person dar, während ein Workflowsystem die Aufgaben dieser Person und ihr Terminkalender deren eigene Planungen enthält. Dabei ist nicht sichtbar, dass die einzelnen Kontextinformationen aus verschiedenen Datenquellen u. U. verschiedene Aspekte des *selben* Kontextobjektes darstellen und daher semantisch zusammengehören (vgl. Abb. 4). Diese Zusammenhänge müssen bisher implizit von der individuellen Clientanwendung oder gar erst vom Benutzer selbst hergestellt werden, was in den bereits angesprochenen komplexen Anwendungsszenarien mit verteilter Entwicklung oder Nutzung von Anwendungskomponenten ebenfalls ein nichttriviales Koordinationsproblem und die Quelle schwer lokalisierbarer Fehler darstellt.

1.2.3 Zielstellung der Arbeit

Wünschenswert wäre ein Verfahren, um sämtliche für einen bestimmten Anwendungsbereich benötigten Kontextinformationen in einer expliziten Form aus allen bekannten Kontextquellen zusammenzutragen, miteinander zu kombinieren, in einer übersichtlichen Form darzustellen und dadurch entsprechenden kontextsensitiven Clientanwendungen ein *vollständiges* (im Sinne von „alle bekannten Informationen sind darin enthalten“), integriertes Kontextmodell präsentieren zu können. Ein solches Kontextmodell erlaubt den Zugriff auf in diversen Quellen befindliche Kontextinformationen mittels einer *realweltbezogenen* Adressierung, d. h. Clientanwendungen werden in die Lage versetzt, sich die von ihnen gewünschten Informationen über ein bestimmtes Realweltobjekt zu beschaffen, ohne sich darüber Gedanken machen zu müssen, aus welchen konkreten Quellen diese Informationen auf welche Weise zu beschaffen sind. Ein solcher expliziter Kontextinformations-Integrationsansatz wird in dieser Arbeit entwickelt und vorgestellt.

Dabei werden die durch ein *Kontextquellenmodell* adressierten Kontextinformationen aus den verschiedenen Kontextquellen in ein separat davon modelliertes und semantisch ausdrucksfähiges *Kontextmodell* einordnet, welches die Realweltobjekte und deren Beziehungen untereinander darstellt, wobei Kontextinformationen als Attribute von Realweltobjekten dargestellt werden. Dadurch wird ein Verbraucher von Kontextinformationen in die Lage versetzt, sich alle zu einem Realweltobjekt gehörenden Informationen durch einen einmaligen Zugriff auf das integrierte Weltmodell zu besorgen, anstatt wie bisher mehrere Kontexterfassungssysteme mittels verschiedener Verfahren befragen zu müssen.

Dazu sind Methodiken notwendig, um Anwendungsdomänen und deren relevante Eigenschaften zu modellieren, Kontextinformationen aus verschiedenen Kontextquellen zu entnehmen und darin einzuflechten. Dabei sollen einzelne Fragmente des Kontextmodells wiederverwendet und in verschiedenen anwendungs- oder problemspezifischen Arten abgebildet werden können. Ebenso soll der Detaillierungsgrad bzw. die Vernetzungsdichte des Modells schrittweise und problemspezifisch verfeinert werden können. Dadurch soll die Erstellung kontextsensitiver Anwendungen und basierend darauf die Durchführung kontextbezogener Abläufe und Tätigkeiten vereinfacht werden. Zusammengefasst ergeben sich die folgenden konkreten Ziele für diese Arbeit:

- Entwicklung eines geeigneten Begriffs- und Beschreibungsmodells zur Modellierung von verteilten Kontextinformationsmengen
- Beschreibung der wichtigsten in diesem Problembereich auftretenden Problemstellungen, Rahmenbedingungen und Anforderungen
- Analyse und Lösung des Verteilungs- und Integrationsproblems durch Entwicklung eines Darstellungs- und Integrationskonzeptes für kontextbezogene Informationen
- Validierung der entwickelten Lösung hinsichtlich der Erfüllung der gestellten Anforderungen, ihrer Realisierbarkeit sowie dem erreichten Mehrwert gegenüber existierenden Lösungen.

Keine Ziele dieser Arbeit stellen die verschiedenen Erfassungsmöglichkeiten für Kontextinformationen, die Entwicklung eines *low-level*-Darstellungsmodells für Kontextinformationen innerhalb einer spezifischen Kontextquelle, die Entwicklung spezieller Zugriffsverfahren auf externe Datenquellen sowie einzelne Interpretations- bzw. Transformationsalgorithmen für Kontextdaten dar.

1.3 Gliederung der Arbeit

Nach der in diesem Kapitel erfolgten grundsätzlichen Vorstellung und Einordnung des Themas in den gesellschaftlichen und wissenschaftlichen Kontext wird dieses in Kap. 2 basierend auf einer Darstellung der zum Verständnis notwendigen Grundlagen auf einer wissenschaftlichen Ebene analysiert. In Kap. 3 wird diese Arbeit durch Darstellung des Bedarfes nach umfassenden kontextbezogenen Informationen im betrieblichen Umfeld auf der wirtschaftlichen Ebene positioniert und motiviert. In Kap. 4 werden existierende Ansätze zur Darstellung und Integration von Kontextinformationen untersucht und basierend auf einer Kritik derselben in Kap. 5 ein neues Modell zur Darstellung integrierter Kontextinformationen entworfen, welches in Kap. 6 anhand repräsentativer Anwendungsbeispiele sowie durch eine Anforderungsanalyse validiert wird. Abschließend wird in Kap. 7 ein Ausblick auf den weiteren Forschungsbedarf auf diesem Gebiet gegeben.

2 Problemanalyse

Das Jahrhundertproblem der Informatik besteht in der Bewältigung des Datenchaos, das infolge historisch, mitunter auch hysterisch und archaisch, sicher aber unkontrolliert gewachsener Datenbestände fast überall entstanden ist.

[Vett89]

Im folgenden werden nach einer Begriffsbestimmung und Darstellung der historischen Entwicklung sowie der wichtigsten verwendeten Technologien auf dem Gebiet der Kontextsensitivität ein Modell zur Beschreibung der Verteilung von Kontextinformationen entwickelt, Arbeitsthesen zur Formalisierung der in Kap. 1 genannten Probleme formuliert sowie ein Modell zur Bewertung entsprechender Lösungen vorgestellt.

2.1 Grundlagen

Aufgrund des fächerübergreifenden Charakters dieser Arbeit werden die verwendeten Konzepte etwas ausführlicher als in einer rein technischen bzw. wirtschaftlichen Arbeit üblich dargestellt.

2.1.1 Begriffsbestimmung

An dieser Stelle werden die wichtigsten in dieser Arbeit verwendeten Begriffe definiert. Speziellere Begriffe werden später im jeweiligen Zusammenhang vorgestellt und definiert.

Realwelt

Unter *Realwelt* wird die Lebenswelt der Menschen, welche sie als „Realität“ ansehen, verstanden. Die Realwelt umfasst neben einem räumlichen Aspekt weitere, etwa soziale oder betriebliche Aspekte, die für einen Nutzer in gleicher Weise relevant und damit „real“ sind. In dieser Realwelt existieren eine Reihe von Objekten (im folgenden *Realweltobjekte* genannt), etwa Personen, Geräte, Termine oder andere Objekte, die für die Tätigkeiten des Nutzers in irgend einer Form relevant sind. Diese relevanten Realweltobjekte gilt es durch *Kontextinformationen* zu beschreiben, wobei unter dem *Kontext* eines Problems der für dieses Problem relevante Ausschnitt aus der Realwelt verstanden wird.

elektronische Welt

Innerhalb von Computersystemen existiert die *elektronische Welt*⁴. In der elektronischen Welt existieren ebenfalls eine Reihe von Objekten, etwa Anwendungen, Daten und elektronische Nachrichten zwischen Anwendungen. Aus der Sichtweise von Anwendungen ist die elektronische Welt „real“ und „substantiell“, während sie die *Realwelt* nicht direkt, sondern nur über Nachrichten von Sensoren erfahren können. Die elektronische Welt erfüllt im Gegensatz zur Realwelt keinen Selbstzweck, sondern dient letztlich zur Unterstützung von Prozessen in der Realwelt (*Realweltprozessen*). Dazu verfügt jede Anwendung

⁴Man könnte sie auch als „Cyberspace“ bezeichnen.

über einen Vorrat an Informationen und Wissen über die Realwelt, der teilweise in Form *kontextbezogener Daten* dargestellt und teilweise implizit in der Anwendungslogik verankert ist. Zumindest Teile der in Datenform dargestellten realweltbezogenen Informationen aus den einzelnen Anwendungen sollen nun aus diesen Anwendungen extrahiert und in gesammelter Form dargestellt werden, wodurch ein umfassendes, vollständiges und anwendungsübergreifendes Bild über die Realwelt erschaffen werden soll.

Kontext

Das Konzept von *Kontext* wird in der wissenschaftlichen Diskussion an vielen Stellen gebraucht, entsprechend verschiedene Ansichten und Definitionen dieses Begriffes existieren (vgl. [Hen⁺02], S. 8 f). Eine Übersicht über Kontextdefinitionen im für diese Arbeit relevanten Bereich der computergestützten Informationsverarbeitung hat DEY in seiner Dissertation zusammengetragen (vgl. [Dey00b, S. 3 ff]). Diese wurden von SPRINGER treffend als „jede für die Charakterisierung eines Entities nutzbare Information“ [Spri04] zusammengefasst. In dieser Arbeit wird unter dem *Kontext* eines Objektes im Speziellen der für dieses Objekt in gewisser Hinsicht relevante Ausschnitt aus der Realwelt verstanden, der durch Computersysteme erfasst und in der elektronischen Welt beschrieben wird. Entsprechend werden unter *Kontextinformationen* eine Menge von in der elektronischen Welt existierenden Informationen verstanden, welche die realweltbezogene Situation eines Realweltobjektes zur Nutzung durch kontextsensitive Anwendungen beschreiben (vgl. [Spri04, S. 8], [Dey00b, S. 4]). Unter *kontextsensitiv* wird dabei ein „System, das die Fähigkeit hat, Aspekte der Umgebung als Kontext zu erfassen und diese für ein kontextbezogenes Verhalten zu nutzen“ verstanden ([ScGe01, S. 2]). Im Gegensatz zu reinen *Umgebungsinformationen* werden unter *Kontext* nur die jeweils *relevanten* Informationen verstanden, d. h. zwei Objekte können in der selben Umgebung unterschiedliche Kontexte haben, wenn sie z. B. unterschiedliche Zielstellungen verfolgen.

Im wirtschaftsinformatischen Begriffsmodell können unter Kontextinformationen konform mit SCHMIDT und GELLERSEN realweltbezogene Informationen, die zusätzlich zu den expliziten Nutzereingaben in Prozesse einfließen, verstanden werden (vgl. [ScGe01]). In diesem Zusammenhang stellen Kontextinformationen eine Teilmenge der *Betriebsinformationen* dar. Während *Betriebsinformationen* sämtliche überhaupt von einem Unternehmen benötigten Informationen umfassen, werden unter *Kontextinformationen* auf die unmittelbare Modellierung der *Realwelt* spezialisierte Betriebsinformationen verstanden, die sich von betriebswirtschaftlichen, etwa monetären oder prozessorientierten Betriebsinformationen durch ihren wesentlich direkteren und abgegrenzteren Bezug zur tatsächlichen Realwelt unterscheiden, d. h. sie stellen nicht wie Betriebsinformationen allgemein bestimmte wirtschaftlich relevante Aspekte der Realwelt dar, sondern die Realwelt selbst. Aus einer wirtschaftlichen Perspektive betrachtet stellt also das auf die Erfassung von Realweltphänomenen spezialisierte Paradigma der Kontextsensitivität aufgrund neuartiger technologischer Möglichkeiten – etwa im Bereich der Sensorik – durch Kontextinformationen *realweltbezogene Betriebsinformationen* bereit, wodurch existierende betriebliche Anwendungen funktionell oder komfortbezogen erweitert sowie neuartige, in besonderem Maß kontextbezogene Unternehmensanwendungen überhaupt erst möglich werden. Einige dieser kontextsensitiven Unternehmensanwendungen werden in Kap. 3 vorgestellt.

Kontextdimension	Beispiele
physikalischer Kontext	Ort, Zeit, Geschwindigkeit von Objekten
Umgebungskontext	umgebende Objekte, Lichtverhältnisse, Geräuschkulisse, Wetter
technischer Kontext	Typ, Eigenschaften und Status der verwendeten und verfügbaren Geräte
persönlicher Kontext	aktuelle Tätigkeit, Gesundheitszustand, Stimmung
sozialer Kontext	in der Nähe befindliche Personen und deren Beziehung zum Nutzer
betrieblicher Kontext	Aufgaben, Rollen und Rechte des Nutzers im betrieblichen Umfeld
Anwendungskontext	Status von Anwendungsprogrammen, E-Mails, besuchte Webseiten

Tabelle 1: Übersicht über die grundlegenden Kontextdimensionen

Name	Inhalt	Bedeutung
gpsLocation	N52° 31.1' E13° 22.57'	Ort des Kontextobjektes in GPS-Koordinaten
velocity	1,8	Geschwindigkeit des Kontextobjektes in m/s

Tabelle 2: Beispiele für Kontextvariablen (vgl. [Mei⁺04, S. 22])

Kontextdimension

Kontext stellt kein homogenes Konzept dar. Kontextinformationen können sehr unterschiedliche, orthogonale Aspekte der Realwelt abbilden, welche im weiteren Verlauf als *Kontextdimensionen* bezeichnet werden (vgl. [Gos⁺03, S. 2 f], [Mei⁺04]). Wie EISENHAUER und KLEMKE richtig feststellen, berücksichtigen die meisten kontextsensitiven Projekte lediglich einzelne Kontextdimensionen (vgl. [Klem00]). Die für fortgeschrittene kontextsensitive Anwendungen benötigten umfassenden Kontextsichten erfordern hingegen eine Integration verschiedener Kontextdimensionen. Tabelle 1 zeigt grundlegende Kontextdimensionen, die in typischen Anwendungsszenarien unterschieden werden können (vgl. [Gos10, S. 7 f], [RaCa03]).

Kontextvariable

Eine *Kontextvariable* gibt einen einzelnen Kontextparameter eines Realweltobjektes an. Analog zu normalen Variablen haben Kontextvariablen einen kontextbezogenen Wertebereich. Beispiele stellen die beiden in Tab. 2 dargestellten Kontextvariablen dar, die ein mobiles Objekt, etwa ein KFZ beschreiben.

Der Kontext eines Realweltobjektes wird durch eine Reihe von Kontextvariablen beschrieben. Beispiele für Kontextvariablen stellen die Temperatur eines bestimmten Raumes oder der Ort einer bestimmten Person dar (vgl. [Lösc02, S. 53 ff], [Mei⁺04, S. 22]). Kontexterfassungssysteme stellen Informationen über den Kontext in Form von Kontextvariablen dar. Das in dieser Arbeit entwickelte Integrationsmodell integriert die von verschiedenen Kontextquellen bereitgestellten Kontextvariablen über Realweltobjekte.

Kontextmodell

Unter einem *Kontextmodell* wird eine strukturierte, menschen- und computerverständliche Rekonstruktion der relevanten Aspekte der Umwelt des jeweiligen Bezugssystems verstanden. Ergänzend zu den vier

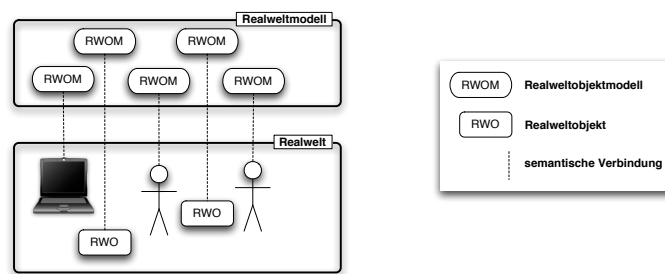


Abbildung 5: Beispiele für semantische Zusammenhänge zwischen realer Welt und deren Repräsentation

hauptsächlich durch die Wirtschaftsinformatik betrachteten Modellarten⁵ bilden Kontextmodelle die relevanten *Realweltobjekte* und deren Eigenschaften ab. Abhängig vom Verwendungszweck können dies allgemein verwendbare räumlich/zeitliche (vgl. [ScTh94, Vol⁺99, Fitz93, Spoh99]) oder mehr anwendungsbezogene Aspekte des Kontextes sein.

Analog zu den Abstraktionsstufen von Informationen⁶ unterscheidet HENRICKSEN verschiedene *Abstraktionsstufen* von Kontextinformationen, konkret *low-level*-Kontextmodelle, welche von Sensoren gelieferte Messwerte darstellen und *high-level*-Kontextmodelle, die aus low-level-Modellen unter Zuhilfenahme von Kontextwissen abgeleitet werden und auch handlungsentscheidende Aspekte beinhalten können (vgl. [Hen⁺02, S. 171]). Beispielsweise kann eine *high-level*-Aussage über das Wetter, etwa das „schönes Wetter“ ist, basierend auf der Auswertung bestimmter *low-level*-Wetterdaten wie Temperatur-, Licht-, Niederschlags-, Wind-, Luftfeuchtigkeits- und Luftdruckwerte formuliert werden. *High-level*-Kontext erlaubt eine einfache, an der Lebenswelt der Nutzer orientierte Verwendung der betreffenden Kontextparameter, was jedoch auf Kosten der Detailgenauigkeit geht (vgl. [FaCl04, S. 1], [Dey⁺01b]). DEY stellt in diesem Zusammenhang die Faustregel auf, dass Kontextinformationen umso einfacher zu benutzen seien, je höher verdichtet sie vorliegen (vgl. [Dey⁺01b, S. 58]).

Unter Kontextmodellen im engeren Sinne werden Daten und Informationen über die Umwelt sowie Informationen darüber, was diese Daten besagen, verstanden. Diese Arbeit verwendet dagegen einen weiter gefassten Kontextmodellbegriff, der zusätzlich modellbezogene Kontextinformationen beinhaltet, d. h. Informationen darüber, woher und wie die Kontextinformationen zu beschaffen sind. Dies bedeutet, dass die in dieser Arbeit entwickelten Kontextmodelle ihren eigenen Kontext beinhalten.

Semantik

Unter der *Semantik* eines Modells wird in dieser Arbeit das Wissen über die Art und Bedeutung der Modellelemente verstanden, d. h. das Wissen, welche Realweltobjekte durch welche Modellelemente abgebildet werden (vgl. Abb. 5, [Morr38], [Schö02, S. 22 f]). Im Gegensatz zur *Pragmatik* beinhaltet dies jedoch nicht das Wissen über die Konsequenzen dieser Information in einer konkreten Situation. Pragmatische Aussagen basieren auf der Kenntnis der Semantik sowie weitreichender Erfahrungen über die in einer bestimmten Situation anzuwendenden Lösungsschritte, die jedoch – da sie sehr anwendungsabhängig sind – nicht Bestandteil des Kontextmodells sein sollen. Solche Informationen werden

⁵Datenmodell, Prozessmodell, Dokumentenmodell und Vorgehensmodell (vgl. [Dres99])

⁶Bei Informationen werden allgemein die Abstraktionsstufen *Zeichen*, *Daten*, *Informationen* und *Wissen* unterschieden (vgl. [ReKr96]).

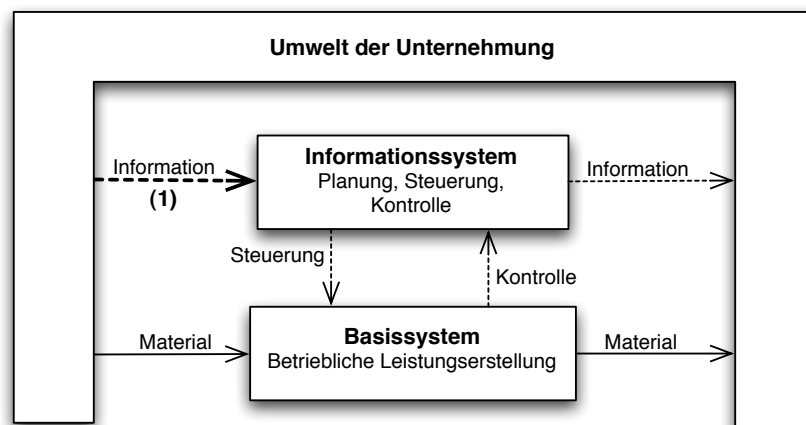


Abbildung 6: Grundmodell der Unternehmung nach GROCHLA ([Groc75])

üblicherweise im Code der Anwendungen, welche das Kontextmodell verwenden oder durch Verfahren des maschinellen Lernens modelliert, z. B. dem fallbasierten Schließen, dessen kontextbezogene Variante *Erfahrungsmanagement* genannt wird (vgl. [Kolo91, Mino01]). Die in dieser Arbeit entwickelten Kontextmodelle müssen also mit solchen Verfahren kombiniert werden, um weitergehende pragmatische Assistenzdienste durch maschinelle Aufgabenträger zu ermöglichen.

2.1.2 Das Unternehmens- und Umweltmodell in der Betriebswirtschaftslehre

Der Begriff der *Umwelt* eines Unternehmens wird in der Betriebswirtschaftslehre an vielen Stellen verwendet. GROCHLA unterscheidet die Unternehmensteile *Basissystem* und *Informationssystem*, wobei das Basissystem für die die Primärprozesse der betriebliche Leistungserstellung (die eigentliche „Produktion“) und das Informationssystem für die Planung, Steuerung und Kontrolle des Basissystems zuständig ist (vgl. [Groc75]). Beide Unternehmensteile stehen mit der *Umwelt* der Unternehmung in Austausch, wobei das Basissystem vor allem die Rohstoffe und Fertigprodukte des Unternehmens und das Informationssystem die für die Produktion relevanten Informationen mit der Umwelt austauscht (vgl. Abb. 6). In diesem Begriffsmodell ist Kontextsensitivität als Teilbereich der mit (1) markierten Beschaffung von Informationen über die Umwelt des Unternehmens einzuordnen. Auch HEINRICH ordnet die *Informationsinfrastruktur* als notwendige Komponente für die Erreichung des Unternehmenserfolges ein, wobei dieser die Beschaffung und Bereitstellung der zur Realisierung der betrieblichen Prozesse notwendigen Informationen obliegt, worunter auch die Beschaffung und Bereitstellung von Kontextinformationen fällt (vgl. [Hein99, S. 20]).

FERSTL und SINZ greifen diesen Modellansatz unter dem Namen *Objektprinzip* auf ordnen ihn in ein generelleres und detaillierteres, *Objektsystem* genanntes Unternehmensmodell ein. Dieses klassifiziert weitere Aspekte von betrieblichen Modellen, u. a. hinsichtlich der Art der Aufgabenträger (persönlich bzw. maschinell), der Art der Beziehungen zwischen Aufgabenträgern (Informationsbeziehung bzw. Nicht-Informationsbeziehung) sowie hinsichtlich der Durchführung bzw. Unterstützung der betrieblichen Leistungsprozesse (vgl. [FeSi98, S. 4 ff]). Sowohl der *Diskurswelt* genannte betriebliche Teil der Realwelt als auch die relevante *Umwelt* des Unternehmens werden in Form von Objekten modelliert. Analog

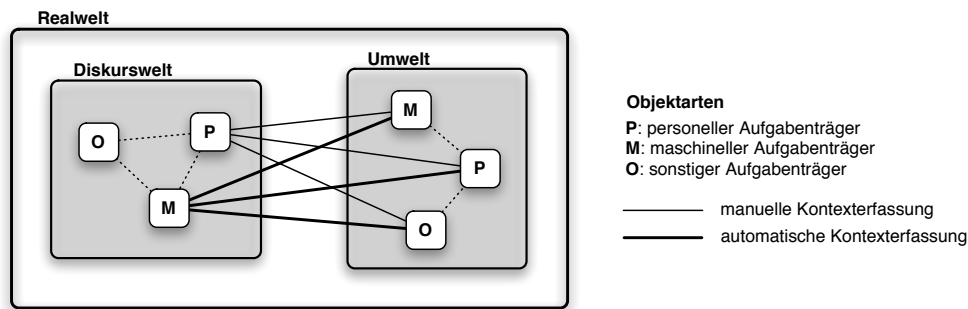


Abbildung 7: Das betriebliche Objektsystem nach FERSTL und SINZ
(vgl. [FeSi98])

werden auch die Beziehungen der Unternehmung mit der Umwelt feingranularer in separate Beziehungen zwischen einzelnen Diskursobjekten und Umweltobjekten untergliedert (vgl. Abb. 7).

Ein in dieser Arbeit thematisiertes kontextbezogenes Informationsmanagement betrachtet vor allem die Informationsbeziehung zwischen dem betrieblichen Informationssystem und der Umwelt der Unternehmung. Abb. 7 stellt die hierbei zu unterscheidenden Beziehungen zwischen Diskurs- und Umweltobjekttypen dar, die in Tab. 3 auf der gegenüberliegenden Seite genauer beschrieben werden. Dabei wird zwischen personellen, maschinellen und anderen Kontextobjekten unterschieden, wobei unter maschinellen Objekten elektronisch kommunikationsfähige Systeme und unter anderen Objekttypen passive, nicht kommunikationsfähige Objekte wie z.B. Gebäude, Kraftfahrzeuge oder Transportcontainer verstanden werden. Diese Arbeit beschäftigt sich vor allem mit der Automatisierung des durch *maschinelle* Diskursobjekttypen realisierten Informationsvorganges über beliebige Umweltobjekttypen, d. h. mit der automatisierten Erfassung von Realweltinformationen durch maschinelle Aufgabenträger im Rahmen von kontextsensitiven Unternehmensanwendungen, was durch die fett gedruckten Verbindungen in Abb. 7 dargestellt wird. Durch diese Automatisierung können Informationen über Realweltobjekte in einer im Vergleich zur manuellen Erfassung wesentlich verbesserten Detailierung, Aktualität, Präzision und Genauigkeit erfasst werden, was neuartige Formen von kontextbezogenen betrieblichen Anwendungen und Gestaltungsmöglichkeiten eröffnet, die in Kap. 3 („Betriebswirtschaftliche Potenziale umfassender Kontextsichtweisen“, S. 49) detaillierter vorgestellt werden.

2.1.3 Historische Entwicklung der Kontextsensitivität

Die technische Entwicklung auf dem Gebiet der Kontextsensitivität verlief in den folgenden Etappen.

1. Die ersten „offiziellen“, d. h. auch so bezeichneten kontextsensitiven Ansätze wurden Anfang der 1990er Jahre von *Want* und wenig später von SCHILIT entwickelt und betrachteten vor allem den Ort von Nutzern und daraus ableitbare Umgebungsinformationen (vgl. [Wan⁺92, ScTh94]). Im Laufe der weiteren Entwicklung wurden schrittweise weitere, vor allem technische Aspekte der Umgebung in Kontextmodelle einbezogen, etwa der Status verfügbarer Geräte und Dienste (vgl. [Fox⁺98, Bru⁺00, Pha⁺01, Jons02]). Diese ersten kontextsensitiven Prototypen beschafften sich ihre Daten komplett selbst, d. h. sie verwalteten sämtliche Sensoren, verteilten und transformierten die von diesen gelieferten Daten und adaptierten sich entsprechend. Diese Lösungen waren, da sie nur

Diskurs-objekttyp	Umwelt-objekttyp	Automatisierungsgrad	Beispiel
personell	personell	manuell	ein Angestellter des Unternehmens besorgt sich Informationen von einer externen Person per Telefonanruf
personell	maschinell	manuell	Abfrage eines fremden Informationssystems durch einen Angestellten des Unternehmens per WWW
personell	anderer Objekttyp	manuell	ein Angestellter schaut nach, wo sich ein bestimmtes Fahrzeug befindet
maschinell	personell	<i>automatisch</i>	eine Unternehmensanwendung bestimmt den Ort eines Kunden des Unternehmens mittels eines Lokalisierungssystems
maschinell	maschinell	<i>automatisch</i>	eine Unternehmensanwendung bestimmt den Status des Mobiltelefons eines Endkunden durch Abfrage eines Web Service des entsprechenden Mobilfunkproviders
maschinell	anderer Objekttyp	<i>automatisch</i>	eine Unternehmensanwendung bestimmt den augenblicklichen Ort eines Transportcontainers durch Abfrage eines Transportsystems

Tabelle 3: Arten von Beziehungen zwischen Diskurs- und Umweltobjekttypen

wenige festgelegte Sensoren verwenden konnten, relativ unflexibel und aufwändig (vgl. [Abo⁺97], [Har⁺03, S. 15 f]).

- Um die Beschaffung von Kontextdaten zu flexibilisieren, wurden später *Kontexterfassungssysteme* entwickelt, welche die von ihnen erfassten Kontextdaten in proprietären Formaten darstellten und verschiedenen Client-Anwendungen zur Verfügung stellten. Dadurch wurde von konkret verwendeten Sensoren abstrahiert. Dazu zählen das von DEY und ABOWD entwickelte *Context Toolkit* (vgl. [Dey⁺99a]), das *Active Map*-System von SCHILIT (vgl. [Schi95]) oder die Arbeiten von BROWN (vgl. [Bro⁺97]).
- Spätere Arbeiten separierten die logischen Kontextmodelle von den konkret verwendeten physikalischen Datenspeicherungsmodellen, um verschiedene kontextbezogene Sichtweisen zu ermöglichen (vgl. [Hong01, Jons02, Har⁺03]).
- Der nächstfolgende Entwicklungsschritt stellt die Integration mehrerer z. T. überlappender, aber getrennter logischer Kontextdatenmodelle in umfassende, anwendungsübergreifende Kontextsichtweisen dar, wie in Abb. 8 auf der nächsten Seite visualisiert. Die vorliegende Arbeit ist auf diesem, noch unzureichend bearbeiteten Entwicklungsschritt anzusiedeln. Arbeiten zu diesem Gebiet werden als verwandte Arbeiten in Kap. 4.6 auf S. 118 vorgestellt.

2.1.4 Kontextquellenarten

Wie bereits angedeutet, werden kontextbezogene Informationen aufgrund der Vielschichtigkeit des für menschliche Nutzer relevanten Kontextes durch verschiedene, auf die Erfassung einzelner Aspekte der

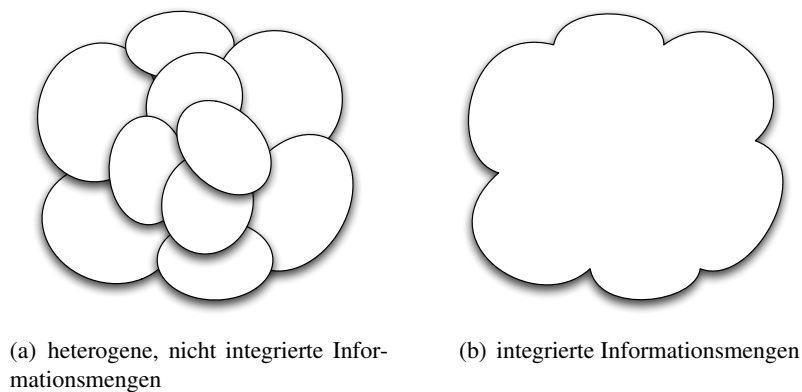


Abbildung 8: Integration heterogener Informationsmengen in umfassende Sichtweisen

Realität spezialisierte Verfahren erfasst und bereitgestellt. Im folgenden werden die wichtigsten Arten von Kontextquellen vorgestellt und kurz charakterisiert.

2.1.4.1 Kontextsensoren

Die Erfassung von physikalischen Merkmalen der Realwelt wird in erster Linie durch *Sensoren* realisiert, welche technische Einrichtungen zur Erfassung von Realweltphänomenen darstellen. Sensoren bestehen in diesem Zusammenhang aus einem Gerät zur Erfassung eines Realweltstimulus sowie einer elektronischen Steuerungseinheit, welche den Sensor mittels elektronisch übermittelter Befehle steuert, d. h. an- und ausschaltet, Messungen initiiert sowie die erhaltenen Messwerte mittels einer Netzwerkschnittstelle in einer maschinenlesbaren, d. h. elektronischen Form bereitstellt (vgl. [Fit⁺02]). Die dabei betrachteten physikalischen Realweltaspekte umfassen jegliche meßbaren, für die Adaption des jeweiligen betrieblichen Leistungsprozesses relevanten Umweltparameter wie Orte, Bewegungs- bzw. Beschleunigungsvektoren (vgl. [Wan⁺92, Wan⁺96, Pri⁺00, Li⁺00, WaAk00, DoZü02, Che⁺99, O.V.04b, AsSt02, DoHu03]); die Identität von Realweltobjekten; Temperatur-, Licht- und Geräuschpegel; magnetische, elektrische oder elektromagnetische Feldstärken in der Umgebung; aber auch physiologische Kenngrößen wie Herzfrequenz, Körpertemperatur oder andere Vitalsignale bzw. die daraus abgeleiteten *high-level*-Aussagen etwa über gegenwärtige Tätigkeit eines Menschen (vgl. [O.V.96b]).

Sensoren sind bereits seit einigen Jahren kommerziell verfügbar und können z. B. in Alltagsgegenstände und -abläufe integriert werden, um Messungen am tatsächlichen Ort des Geschehens, d. h. der Lebenswelt der menschlichen Nutzer, durchzuführen. Als Beispiel seien die *iButton*[®]-Baureihe der Firma DALLAS SEMICONDUCTOR oder die Produkte der Firma UBISENSE genannt (vgl. [O.V.04e, O.V.04a]). SCHMIDT gibt in [Schm02, S. 45 ff] eine Übersicht über Sensortechnologien. Eine Sonderform von Sensoren stellt das von KAHN, KATZ und PISTER vorgestellte Konzept des *Smart Dust* dar, bei dem viele winzige, mit einer drahtlosen Verbindungsmöglichkeit ausgestattete Sensoren wie Staub über einem beliebigen Gebiet verteilt werden und sich über peer-to-peer-Kommunikation zu spontanen, selbstorganisierten Sensornetzwerken verbinden (vgl. [Kah⁺99]). Über eine Integration von *Smart Dust* in Tapete, Wandfarbe, Asphalt oder Kleidung lassen sich bei Bedarf kostengünstig großflächige Gebiete

mit Sensortechnik ausstatten und für kontextbezogene Dienste verfügbar machen, falls diese Form der Überwachung überhaupt sinnvoll, gesellschaftlich gewünscht bzw. im gesetzlichen Rahmen erlaubt ist.

Daneben werden in einigen Forschungsprojekten die auch als *logische Sensoren* bezeichneten *Software-Sensoren* zum Zugriff auf bereits gespeicherte Informationen in anderen Anwendungen verwendet (vgl. [Sch⁺99a, S. 100 f]). Dies geschieht unter dem Paradigma, dass Anwendungen diese Informationen freiwillig mit anderen Anwendungen teilen und die notwendigen Zugriffsrechte berücksichtigt werden.

Das Gegenstück zu Kontextsensoren stellen *Kontextaktoren* dar. Darunter werden vernetzte Geräte verstanden, die durch elektronisch übermittelte Steuerungsbefehle Aktionen in der Realwelt durchführen. Prinzipiell können sämtliche durch elektronisch vernetzte Maschinen realisierte Tätigkeiten als Kontextaktionen aufgefasst werden. Das Spektrum der Aktionen reicht von einfachen Ausdrucken von elektronisch übermittelten Informationen auf (physikalischem) Papier durch einen netzwerkfähigen Drucker über die Steuerung z. B. der Lichtverhältnisse durch vernetzte Lichtschalter bis hin zur Erledigung komplexer Tätigkeiten durch drahtlos vernetzte Roboter.

2.1.4.2 Kontexterfassungssysteme

Damit Kontext nicht mehr von jeder kontextsensitiven Anwendung separat und redundant erfasst werden muß, sondern alle Anwendung komfortabel über einheitliche Schnittstellen auf ganze Klassen von Kontextsensoren und -aktoren zugreifen können, ohne sich über die technischen Fragestellungen der Energieversorgung, Vernetzung und Steuerung der Sensoren und Aktoren Gedanken machen zu müssen, wurden *Kontexterfassungssysteme* entwickelt. POWER bezeichnet diese als „*software system whose task is to perform the gathering and management of context information from the (...) environment and make this information available to consumers of context*“ [Powe03, S. 3]. Diese Infrastrukturen bestehen aus einem *Sensordienst*, der Kontext mittels Sensoren bzw. Aktoren erfasst und verändert, einer *Verarbeitungseinheit*, welche die von den Sensoren gelieferten Rohdaten von Meßfehlern bereinigt, verdichtet, grundlegend interpretiert sowie miteinander kombiniert und daraus Kontextvariablen generiert, welche in einem *Kontextmodell* repräsentiert und ggf. in einer *Kontextdatenbank* persistent gespeichert werden, um letztlich mittels einer *Anfrageschnittstelle* durch Kontextverbraucher zugegriffen zu werden (vgl. Abb. 9(a) auf der nächsten Seite, [Spri04, S. 9]).

Bekanntester Vertreter ist das *Context Toolkit* von DEY und ABOWD (vgl. [Dey⁺99a, Dey00b]) sowie dessen vielfältige Erweiterungen, z. B. durch JANG (vgl. [Jan⁺01]). Weitere Frameworks stellen *stick-e-notes* von PASCOE (vgl. [Pasc97]), der *SitComp*-Dienst von HULL (vgl. [Hul⁺97]), das *Context Framework* von LÖSCHAU (vgl. [Lösc02]), *Context Shadow* von JONSSON (vgl. [Jons02]), *Gaia* von CAMPBELL (vgl. [Rom⁺02]), das *TEA*-Projekt von SCHMIDT (vgl. [Sch⁺99a]) sowie *Context Fabric* von HONG dar (vgl. [Hong01]).

Typischerweise stellen Kontexterfassungssysteme einfache Kontextmodelle über eine Schnittstelle (API) bereit, die zwar aus mit den entsprechenden Realweltobjektmodellen assoziierten Kontextvariablen bestehen können, jedoch nicht vollständig im Sinne von Kap. 1.2.3 („Zielstellung der Arbeit“, S. 9) sind. Dies liegt – wie später noch ausführlicher dargestellt wird – daran, dass Kontexterfassungssysteme aus Komplexitätsgründen auf die Erfassung einzelner Kontextdimensionen und damit nur einzelner Aspekte von Realweltobjekten spezialisiert werden können. Sie strukturieren die ihnen bekannten Kon-

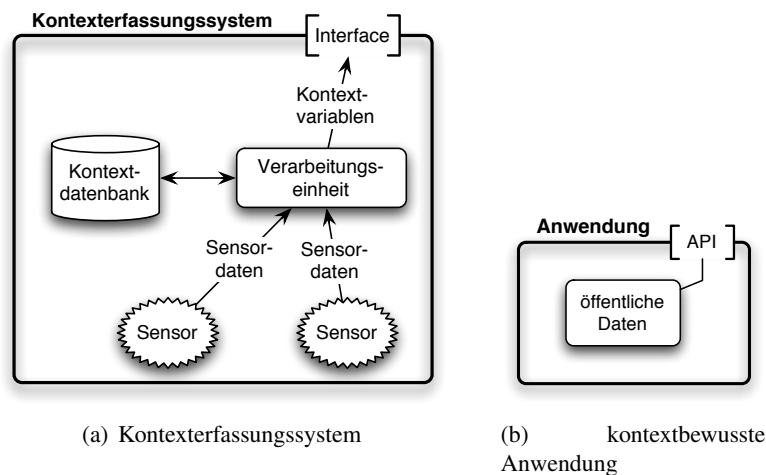


Abbildung 9: Schematischer Aufbau von Kontextquellen

textinformationen prinzipbedingt mittels einfacher, performanter, technischer Datenmodellierungsmethoden wie Variablen, Objektstrukturen oder Datensätzen in Datenbanken. Häufig sind die dabei verwendeten Verfahren auf die Auswertung der mit den gegebenen Sensoren erfassten Kontextdaten und nicht auf die effektive Kombination mit weiteren, andersweitig gewonnenen Kontextinformationen spezialisiert (vgl. [Hul⁺97, S. 9]).

2.1.4.3 Kontextbewusste Anwendungen

Informationen über den Kontext müssen nicht ausschließlich von Kontexterfassungssystemen stammen. Prinzipiell stellt jede Computeranwendung ein mentales Werkzeug des Menschen zur Unterstützung eines Vorganges in der Realwelt dar und enthält deswegen Informationen über diese Realweltdomäne und damit möglicherweise über die darin enthaltenen Realweltobjekte, welches von anderen Anwendungen kontextbezogen wiederverwendet werden könnte (vgl. [Sch⁺99a], [Gos⁺03, S. 3 f]). Es wäre daher sinnvoll, wenn ein Anwendungssystem bestimmte Informationen ebenfalls über eine Schnittstelle verfügbar macht, wenn diese Informationen in einem anwendungsübergreifenden Zusammenhang Verwendung finden sollen (vgl. Abb. 9(b)).

Beispiel 2 *Der Terminkalender eines Nutzers enthält Informationen über dessen geplante Termine sowie u. U. Informationen über darin involvierte Personen oder Orte. Das Adressbuch enthält in gleicher Weise strukturierte Informationen über die Bekannten des Nutzers.*

Einzuhaltende Termine des Nutzers stellen für viele Anwendungen, etwa ein später noch detaillierter besprochenes KFZ-Navigationssystem handlungsentscheidende Kontextinformationen dar. Auch die Kenntnis der Bekannten und Kollegen des Nutzers kann in einem solchen Zusammenhang von Nutzen sein, etwa um diese Personen anhand verschiedener im Adressbuch enthaltener synonyme Namen zu identifizieren. Anzumerken bleibt, dass die in Beispiel 2 dargestellten Kontextinformationen ausschließlich in diesen Anwendungen enthalten sind und daher andersweitig nur sehr aufwändig oder überhaupt nicht zu beschaffen wären.

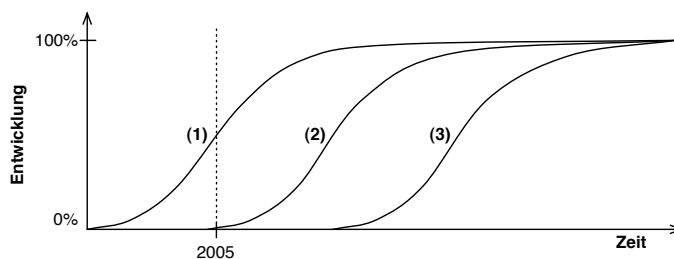


Abbildung 10: Entwicklungswellen von kontextbezogener Technologie

2.2 Arbeitsthesen

Als Arbeitsthesen für die vorliegende Arbeit werden die folgenden Aussagen formuliert, welche im weiteren Verlauf dieser Arbeit validiert werden.

These 1 *Kontextinformationen entstehen prinzipbedingt verteilt in verschiedenen Kontextquellen.*

Ausgangspunkt dieser Arbeit ist die Annahme, dass Kontextinformationen durch verschiedene Kontextfassungssysteme bereitgestellt werden, woraus sich der in Kap. 1.2.2 („Problemstellung“, S. 8) beschriebene Integrationsbedarf für verteilte und heterogene, d. h. in verschiedenen Kontextquellen enthaltene Kontextinformationen überhaupt erst ergibt. Als Ursachen für die Existenz heterogener Kontextquellen können die im folgenden detaillierter dargestellten Faktoren identifiziert werden, welche die prinzipielle Natur einer heterogenen Kontextquellenlandschaft belegen.

Entwicklung in Wellen. Die Einführung kontextsensitiver Verfahren erfordert wie jede grundlegende technologische Innovation Investitionen. Aufgrund der nicht unerheblichen Gesamtkomplexität wirkt sich das dabei in der Praxis außerordentlich relevante Verhältnis von Kosten und Nutzen problematisch auf die Einführung dieser kontextverarbeitenden Systemen aus, denn die dafür notwendigen Investitionen, welche für die Realisierung von Infrastrukturen zur Erfassung, Auswertung und Kombination von Kontextinformationen, die Einbettung von kontextbezogenen Routinen in Informationssysteme, die Anpassung der betroffenen Anwendungen und Geschäftsprozesse, die Fortbildung von Nutzern, zur Behebung anfänglicher Fehler sowie zur Überwindung der in diesem Zusammenhang notwendigen gesellschaftlichen Paradigmenwechsel⁷ anfallen, sind bei kontextbezogenen Technologien erheblich und fallen an, bevor die entsprechenden Verfahren einen Nutzen realisieren.

Die Einführung von kontextverarbeitenden Systemen wird daher absehbar nicht in einem Stück, sondern in mindestens drei aufeinanderfolgenden Wellen erfolgen (vgl. Abb. 10, [Hul⁺97, S. 12]). Als *erste Welle* werden vor allem kontextsensitive Insellösungen mit relativ geringem Implementationsaufwand und hohem Nutzen realisiert. Dazu zählen auf bereits existierenden zellulären Mobilfunknetzen basierende ortsbezogene Dienste wie das WOMBAT-System (vgl. [Fra⁺01, DoHu03, O.V.04b]), die Optimierung der Logistik mittels durch Inhouse-Lokalisierungssystemen optimierten betriebswirtschaftlichen Logistikprozessen, wie von Müller am Beispiel der Warteschlangen von Patienten an verschiedenen Stationen im Krankenhaus demonstriert (vgl. [Müll03]) oder kontextsensitiv aufgewertete Verkaufsfrentends wie

⁷In diesem Zusammenhang sei an den anfänglichen und teilweise durchaus eskalierenden gesellschaftlichen Widerstand gegen die Einführung der weniger brisanten mechanischen Produktionsanlagen wie Webmaschinen oder die Eisenbahn erinnert.

der *Future Store* der METRO GROUP, welcher den Kunden u. a. zu gewünschten Regalstandorten navigiert (vgl. [Grou03]). Erst wenn diese sich selbst amortisierenden kontextsensitiven Systeme existieren, werden darauf aufbauend als *zweite Welle* schrittweise weitere stark nutzbringende kontextbezogene Dienste folgen, die jedoch aufgrund ihres eigentlich zu hohen Investitionsbedarfes erst durch die Erweiterung eines in der ersten Welle entwickelten und bereits amortisierten Kontextdienstes ökonomisch sinnvoll bzw. technisch möglich werden. Schließlich lohnt bei Vorhandensein einer gewissen Menge von *teilweise* vernetzten Kontextquellen die in dieser Arbeit beschriebene Vernetzung *aller* verschiedenen Kontextquellen unter der Perspektive eines ganzheitlichen Informationsmanagements, wodurch schließlich die *dritte Welle* von kontextbezogenen Diensten, welche Synergien existierender Kontextdienste nutzen, entsteht.

Spezialisierung auf Erfassung von einzelnen Kontextdimensionen. Da der relevante Kontext möglicherweise sehr heterogen ist, können die Anforderungen zur Erfassung verschiedener Kontextdimensionen durchaus widersprüchliche Anforderungen beinhalten, weswegen einzelne Kontextdimensionen nur durch spezialisierte Infrastrukturen in der benötigten performanten, ergonomischen bzw. ökonomisch sinnvollen Form erfasst werden können. Unterschiede können sich etwa hinsichtlich der Gestaltung und Einbettung von Sensoren in die Realwelt, den verwendeten Infrastrukturen, Kommunikationsprotokollen und Sicherheitsverfahren, der Art und Auslegung von Komponenten hinsichtlich Leistungsfähigkeit und Kosteneffizienz, der Nutzung ggf. lizenzpflichtiger Auswertungsverfahren sowie diversen spezialisierten Datenformaten ergeben. Da es also u. U. unwahrscheinlich bzw. sogar unmöglich sein kann, sämtliche Anforderungen durch ein einziges Kontexterfassungssystem zu befriedigen, da dieses zu komplex und unspezifisch ausfallen würde, müssen bei Bedarf mehrere, auf verschiedene Anforderungsbereiche spezialisierte Kontexterfassungssysteme miteinander kombiniert werden. Ein Beispiel stellt die prinzipiell unterschiedliche Herangehensweise bei der Lokalisierung von Endgeräten innerhalb und außerhalb von Gebäuden dar. Innerhalb von Gebäuden wird diese mit einer sehr großen Genauigkeit über stationäre Funkbaken und außerhalb von Gebäuden, wo es vor allem auf eine große Reichweite und weniger auf eine Lokalisierung im Millimeterbereich ankommt, über ein globales Satellitensystem realisiert. Keines der Systeme erfüllt die Anforderungen im anderen Einsatzbereich bzw. funktioniert dort überhaupt. Verschlimmert wird dieses Problem durch die Nutzung von Systemen verschiedener Hersteller, welche zueinander inkompatible Datenformate verwenden. Die von solcherart heterogenen Kontextquellen gelieferten Kontextinformationen müssen daher in einem separaten Integrationsschritt in eine umfassende Sichtweise auf die Realwelt integriert werden.

Redundante Erfassungsverfahren. Für den praktischen Einsatz werden *robuste* und dauerhaft, d. h. auch in verschiedenen Situationen und Umgebungen verfügbare Kontextinformationen auf einem hohen Qualitäts- und Abstraktionsniveau benötigt (vgl. [Hong01, S. 1]), wobei die Eignung verschiedener Kontexterfassungsarten in unterschiedlichen Einsatzbereichen wie eben bereits dargestellt durchaus variabel ist. Um eine angemessene Verfügbarkeit und Qualität der Kontextinformationen zu erreichen, müssen daher ggf. verschiedene Verfahren, die in unterschiedlichen Situationen unterschiedlich geeignet sind, kombiniert werden (siehe dazu auch Kap. 3.1 („Modellseitige Effekte durch die Kombination von Kontextinformationen“, S. 49)). Bei sehr ähnlichen Verfahren kann dies bereits in den Kontextquellen erfolgen, was jedoch nicht immer der Fall sein muß. In letzteren Fällen ergibt sich die Notwendigkeit,

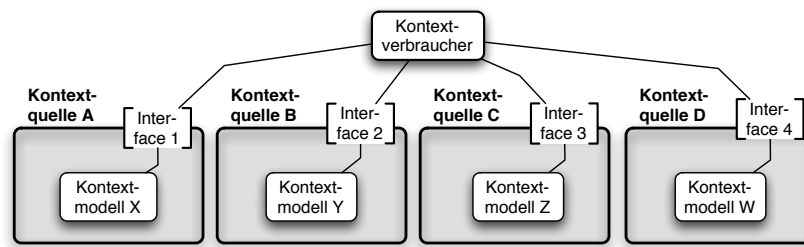


Abbildung 11: Heterogene Kontextquellen

die von verschiedenen Kontextquellen gelieferten Kontextinformationen auf einer separaten Ebene zu integrieren.

Beteiligung von Anwendungsdaten. Ein weiterer Grund für die Annahme einer Mehrzahl von Kontextquellen resultiert aus der bereits angesprochenen Nutzbarkeit von bestimmten Anwendungsinformationen in einem kontextbezogenen Zusammenhang, weshalb auch normale Anwendungen als Kontextquellen angesehen werden können. Diese stellen prinzipbedingt verteilte und heterogene Kontextquellen dar, die mit den durch ein oder mehrere Kontexterfassungssysteme bereitgestellten Kontextinformationen zu kombinieren sind.

Letztlich können also eine Reihe von unterschiedlichen *Kontextquellen* unterschieden werden, deren Informationen miteinander zu kombinieren sind, entweder um die *Breite* der Kontextdatenbasis durch eine Integration unterschiedlicher, separat erfasster Kontextdimensionen zu erhöhen, oder um deren *Tiefe*, *Verfügbarkeit* oder *Datenqualität* (Genauigkeit, Aktualität, Zuverlässigkeit) durch die Kombination von gleichartigen Meßergebnissen zu verbessern. Dadurch ergibt sich die für diese Arbeit ausschlaggebende Situation, dass eine kontextsensitive Anwendung die benötigten Kontextinformationen aus mehreren Kontextquellen besorgen muß (vgl. Abb. 11).

These 2 *Ein umfassendes Bild über die Umwelt kann in betrieblichen Anwendungsszenarien einen Nutzen stiften.*

Diese These wird in Kap. 3 („Betriebswirtschaftliche Potenziale umfassender Kontextsichtweisen“, S. 49) analysiert und insofern validiert, indem die wichtigsten Anwendungsfelder von umfassendem Kontextwissen in betrieblichen Anwendungssystemen vorgestellt und die diesbezüglichen technischen, wirtschaftsinformatischen und wirtschaftlichen Effekte skizziert werden.

These 3 *Es ist möglich und sinnvoll, Kontextinformationen unterschiedlicher Herkunft und Form in ein gemeinsames Weltmodell einzubinden.*

Diese These basiert auf der durch These 1 beschriebenen Ausgangssituation und wird durch die in These 2 beschriebenen ökonomischen Potenziale integrierter Kontextinformationen motiviert. Die Einbindung heterogener Kontextinformationen in ein umfassendes Weltmodell sollte möglich sein, da sich sämtliche Kontextinformationen letztendlich alle auf eine einzige, widerspruchsfreie Realwelt beziehen. Widersprüchliche Kontextinformationen finden ihre Ursache letztlich immer in veralteten oder zu groben Ansichten der Realität und können entsprechend durch eine feingranularere Modellierung aufgelöst werden.

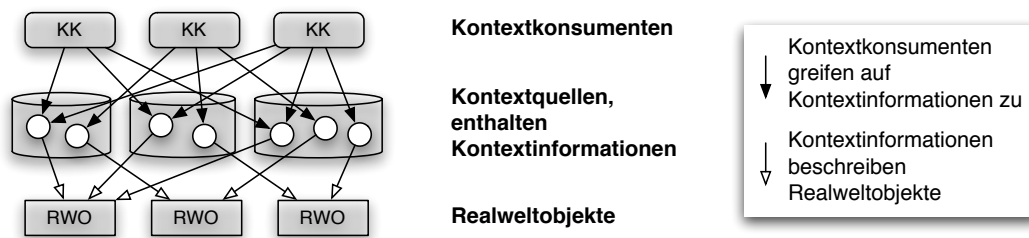


Abbildung 12: Verteilungsmodell für Kontextinformationen

Diese These wird in Kap. 6 („Validierung“, S. 175) durch die praktische Anwendung des dazu in Kap. 5 („Context-Maps“, S. 135) entwickelten Darstellungs- und Integrationsmodells für kontextbezogene Informationen durch die Modellierung typischer Anwendungsfälle validiert.

2.3 Verteilungsmodell für Kontextinformationen

Nachdem die Ausgangslage dieser Arbeit beschrieben und begründet wurde, wird an dieser Stelle ein Verteilungsmodell für verteilte und heterogene Kontextinformationen entwickelt, welches die formale Grundlage der weiteren Arbeitsschritte auf diesem Gebiet darstellt. Sämtliche Anwendungen, die kontextbezogene Informationen liefern können, d. h. alle Arten von Kontexterfassungssystemen und kontextbewusste Anwendungen werden im folgenden als *Kontextquellen* aufgefasst und unabhängig von ihrer internen Struktur und Arbeitsweise in gleicher Weise als solche behandelt. Jede Kontextquelle enthält Kontextinformationen in Form von *Kontextvariablen*, welche jeweils einen einzelnen Aspekt eines Realweltobjektes beschreiben. Dabei können verschiedene Kontextvariablen in verschiedenen Kontextquellen verschiedene Aspekte eines Realweltobjektes beschreiben, sogenannte *synonyme* Kontextvariablen in verschiedenen Kontextquellen sogar den selben Aspekt des selben Realweltobjektes.

Die Daten sollen dabei nicht aus den Kontextquellen „gestohlen“, sondern auf eine vorgesehene Weise unter Nutzung einer expliziten Zugriffsschnittstelle unter Berücksichtigung von Zugriffsrechten und anderen Rahmenbedingungen aus den Kontextquellen importiert werden. Dazu stellen sämtliche zu verwendenden Kontextquellen die von ihnen gelieferten kontextbezogenen Informationen mittels einer wie auch immer gearteten Schnittstelle (engl. *Interface*) zur Verfügung (vgl. Abb. 11 auf der vorherigen Seite). Dabei treten zwangsläufig mehrere Arten von Schnittstellen auf, die in gleicher Weise verwendet werden sollen.

Auf der Seite der Kontextnutzung können ebenfalls verschiedene menschliche Nutzer oder maschinelle Anwendungen identifiziert werden, die Kontextinformationen zugreifen, um sie für eine kontextsensitive Funktionalität zu verwenden. Diese werden im folgenden ebenfalls unabhängig von ihrer Art, internen Struktur und Arbeitsweise als *Kontextkonsumenten* bezeichnet. Abb. 12 stellt die resultierende Situation dar.

Weitergehende Aspekte wie ein Suchdienst für Kontextquellen, welcher geeignete Kontextquellen sucht und mitsamt einer Beschreibung der enthaltenen Kontextvariablen und benötigten Zugriffsverfahren bereitstellt, sind hierbei im weiteren Sinne durchaus von Bedeutung, spielen für die konkrete Themenstellung jedoch keine entscheidende Rolle und werden deshalb im weiteren Verlauf der Arbeit

ausgeklammert und in den Ausblick verschoben. Ebenfalls irrelevant für diese Arbeit ist die Entwicklung spezieller Zugriffsverfahren für diverse Schnittstellenarten.

2.4 Anforderungsanalyse

Ein geeignetes Kontextinformationsintegrations- und -darstellungsmodell muß einer Reihe von Anforderungen genügen, auf die im folgenden – untergliedert in einzelne Anforderungsbereiche – eingegangen wird. Auf dieser Anforderungsanalyse basiert das zum Vergleich der Tauglichkeit der in Kap. 4 („Verwandte Arbeiten“, S. 97) und Kap. 5 („Context-Maps“, S. 135) vorgestellten Lösungsansätze verwendete Bewertungsschema.

2.4.1 Bewertungsschema

Um die verschiedenen Lösungsansätze objektiv hinsichtlich der relativ komplexen Zielstellung dieser Arbeit vergleichen zu können, wird diese Zielstellung in eine Reihe von grundlegenden *Anforderungskomplexen* untergliedert, welche aus einzelnen, separat zu analysierenden *Anforderungen* bestehen. Um die Analyse der einzelnen Anforderungen zu objektivieren, werden für jede Anforderung eine Reihe von *Kriterien* definiert, welche durch den betrachteten Lösungsansatz erfüllt werden müssen.

Die Auswahl der verwendeten Anforderungen und Kriterien basiert zum einen auf der Sammlung der in der Kontext-Community definierten Anforderungen, welche vom Autor dann aufgrund seiner jahrelangen Erfahrungen mit kontextsensitiven Anwendungen und Forschungsprojekten vervollständigt und in [Gosl04] durch Publikation validiert wurden.

Eine *Gewichtung* einzelner Kriterien hängt sehr stark von den konkreten Rahmenbedingungen im jeweiligen Anwendungsfall ab. Da es in dieser Arbeit nur um den *Vergleich* der Funktionalität und nicht um eine absolute *Bewertung* der dargestellten Lösungen hinsichtlich eines konkreten Anwendungsfalles geht, wurde auf eine Gewichtung der einzelnen Bewertungskriterien und -anforderungen verzichtet und lediglich überprüft, ob die betreffenden Kriterien erfüllt sind, wobei eine möglichst vollständige Unterstützung aller Kriterien gefordert ist. Dabei wird das in Tab. 4 auf der nächsten Seite dargestellte Bewertungsschema verwendet. Die Bewertung von Anforderungskomplexen stellt den arithmetischen Durchschnitt der Bewertungen der einzelnen Anforderungen im betreffenden Anforderungskomplex dar.

Im folgenden werden nun die einzelnen Anforderungen an Modelle zur Integration und Darstellung kontextbezogener Informationen, d. h. zur Bewertung verwandter Arbeiten als auch der in dieser Arbeit entwickelten Lösung detaillierter vorgestellt.

2.4.2 Allgemeine Verwendbarkeit

An dieser Stelle werden allgemeine Kriterien für eine gute Handhabbarkeit und Verwendbarkeit eines Kontextmodells vorgestellt.

Anforderung 1 – *Allgemeine Verwendbarkeit*

Speziell ergeben sich in diesem Bereich die im folgenden dargestellten Anforderungen.

Anforderung 1a – *Formalisierungsgrad*

Erfüllungsgrad	Pkt	Anzahl erfüllter Kriterien
Anforderung erfüllt		
Anforderung voll erfüllt	5	alle Kriterien erfüllt
Anforderung weitgehend erfüllt	4	ein Kriterium wurde nicht erfüllt
Anforderung eingeschränkt erfüllt	3	die meisten Kriterien wurden grundlegend erfüllt, es fehlen jedoch mehrere Kriterien
Anforderung nicht erfüllt		
Anforderung nur proprietär erfüllbar	2	die Lösung kann die Anforderung in ihrer standardisierten Form nicht erfüllen, jedoch relativ einfach hinsichtlich der geforderten Funktionalitäten erweitert werden.
Anforderung nicht erfüllbar	1	die Lösung kann die Anforderung nicht erfüllen, zur Erweiterung der Lösung hinsichtlich der geforderten Kriterien wären komplexe Änderungen der dargestellten Technologie notwendig.

Tabelle 4: Schema zur Bewertung von Lösungsansätzen hinsichtlich der Integration von heterogenen Kontextinformationen

Um ein eindeutiges, anwendungs- und nutzerübergreifendes Verständnis, automatisierbare Auswertungen sowie Konvertierungen der enthaltenen Informationen in andere Repräsentationen oder Ontologien zu ermöglichen, müssen sämtliche Elemente des Kontextmodells hinreichend formal modelliert sein (vgl. [Hen⁺02, S. 172]). Auch die im folgenden noch eingehender diskutierte Zusammensetzung der Datenbasis aus separaten Teilmodellen kann nur auf einer klaren formalen Spezifikation aller enthaltenen Elemente basieren. Daher sollen sämtliche im Modell verwendeten Elemente in einer Ontologie formalisiert sein. Im einzelnen werden dabei die folgenden Kriterien gefordert:

- Möglichkeit zur formalen Beschreibung des Modells mittels einer Ontologie,
- Hierarchischer Aufbau der Ontologie zur Spezifikation von Generalisierungsbeziehungen,
- Formalisierung sämtlicher Modellelemente: Realweltobjekte, Beziehungen zwischen diesen Objekten, Kontextvariablen als Attribute von Realweltobjektmodellen.

Anforderung 1b – Modularität

Um die auftretende Komplexität der teilweise sehr detaillierten und umfangreichen Kontextmodelle angemessen handzuhaben, sollen die einzelnen Kontextmodelle modular aus miteinander verknüpften und wiederverwendbaren Teilmodellen zusammengestellt werden können. Einzelne Teilmodelle sollen dabei von verschiedenen Anwendungen definiert und genutzt werden können. Im einzelnen sind diese Kriterien zu erfüllen:

- Möglichkeit zur Darstellung separater Teilmodelle,
- Möglichkeit zur Adressierung von Modellelementen in anderen Modulen,

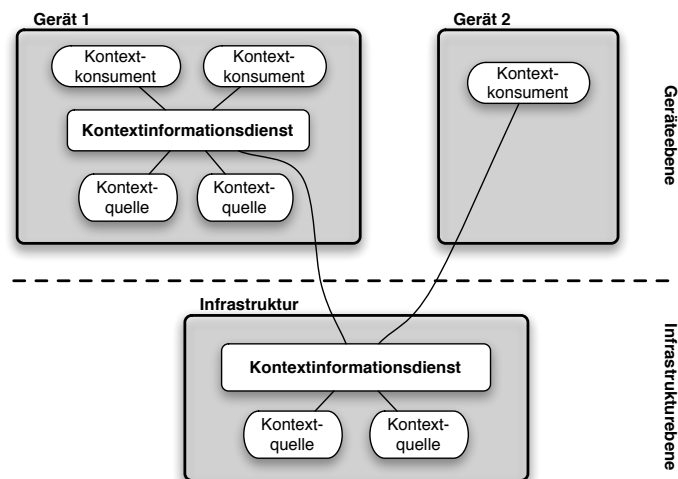


Abbildung 13: Vernetzung der Dienste von Endgeräten und Infrastruktur

- Möglichkeit zur Kombination von Teilmodellen zu größeren Einheiten.

Anforderung 1c *Verteilte Darstellung und Speicherung, Replizierbarkeit*

Kontextsensitive Infrastrukturen bestehen aus einer Vielzahl von in die Realwelt eingebetteten mobilen und stationären Geräten, auf denen jeweils verschiedene Kontextquellen und -verbraucher laufen. In gleicher Weise gibt es eine stationäre Infrastruktur, welche die selben Eigenschaften aufweist. Damit das von den verschiedenen Geräten gesammelte Kontextmodell auf *allen* Endgeräten zur Nutzung durch die dort laufenden Kontextverbraucher verfügbar ist, muß ein Kontextinformationsdienst die auf verschiedenen Endgeräten bereitgestellten und nachgefragten Kontextinformationen transparent zwischen den einzelnen Endgeräten austauschen (vgl. Abb. 13, [Hong01, S. 6], [Kadn04a, S. 48 ff]). Das *Evernet*, d. h. ausreichend schnell, permanent und überall verbundene Endgeräte, ist jedoch nach wie vor eine Vision (vgl. [Bers00]). In der Praxis können Verbindungen zu mobilen Endgeräten aufgrund ökonomischer als auch technischer Restriktionen nicht permanent aufrecht erhalten werden, denn die Verwendung populärer drahtloser Kommunikationsnetzwerke wie UMTS, GSM oder WLAN verursacht Kosten und verkürzt die Betriebszeit batteriebetriebener mobiler Endgeräte signifikant, außerdem erfolgt die Anbindung nicht immer in der benötigten Geschwindigkeit etwa hinsichtlich der Antwortzeiten entfernter Geräte.

Daher müssen sämtliche zu verwendenden Inhalte, d. h. auch das Kontextmodell bzw. Teile davon, auf den mobilen Geräten zwischengespeichert werden können, was durch geeignete Replikationsmechanismen erfolgen sollte. Für eine funktionsfähige, d. h. auch im abgekoppelten Zustand vollständig benutzbare Replik, welche die Einbettung der auch im abgekoppelten Zustand immer noch lokal auf dem jeweiligen Gerät gewonnenen Kontextvariablen in das zwischengespeicherte anwendungs- und geräteübergreifende Kontextmodell erlaubt, werden neben den eigentlichen Kontextdaten auch diverse Metadaten benötigt, etwa semantische Informationen über die Bedeutung und Herkunft der externen Daten, um diese angemessen mittels ebenfalls zu replizierenden Interpretations-, Transformations- und Kombinationsalgorithmen mit den lokal gewonnenen Kontextvariablen zu kombinieren. Diese Replikationsstrategie

kann auch zur Speicherung bereits abgerufener Daten (Caching) oder durch Vorabladen von zukünftig benötigten Daten (Prefetching) auch zur Performancesteigerung bzw. Transferkostensenkung verwendet werden und ist damit im praktischen Einsatz durchaus in verschiedener Hinsicht relevant. Es ergeben sich die folgenden Kriterien:

- Unterteilbarkeit des Modells in separat replizierbare Teile,
- Replizierbare Darstellung von Daten
- Replizierbare Darstellung von Methoden, um die im abgekoppelten Zustand auf den Endgeräten gewonnenen Daten in die Replik zu integrieren

Anforderung 1d – *Unabhängigkeit von Programmiersprachen und -plattformen*

Im Unternehmensumfeld werden durchaus verschiedene Programmiersprachen verwendet. Klassische Unternehmensanwendungen sind in *Cobol*, moderne in *Java* oder *C++* verfasst. Auch im mobilen Sektor werden trotz der Dominanz von *Java* durchaus verschiedene, häufig hardwarenähere Sprachen eingesetzt. Ähnliches gilt für die verwendeten Computerplattformen. Neben den im Unternehmensumfeld nach wie vor dominierenden *UNIX*-artigen Betriebssystemen haben sich mit *Linux*, *Windows* und *OS X* mindestens drei weitere Plattformen etabliert, deren Marktanteile derzeit steigen. Da Kontextinformationen durch sämtliche Anwendungen benutzbar sein sollten, muß die Kontextinformationsbasis also konzeptionell möglichst plattform- und programmiersprachenunabhängig sein, was durch die folgenden Anforderungen formalisiert wird.

- datenbezogener Charakter der Kontextinformationsbasis
- programmiersprachenspezifische Elemente wie im Modell verwendete Algorithmen und Codeteile dürfen nur innerhalb des Modells verwendet werden

Anforderung 1e – *Flexible Adressierungsmöglichkeiten*

In der klassischen Datenhaltung sind die zugegriffenen Daten ausgehend von wohlstrukturierten Anwendungsdomänen und Datenbeständen oft eindeutig bekannt und können direkt adressiert und zugegriffen werden. Im kontextsensitiven Szenario sind die gesuchten Realweltobjekte wesentlich häufiger nicht direkt bekannt, sondern werden erst durch eine Kombination von kontextbezogenen Aussagen bestimmt.

Beispiel 3 *Bei der Steuerung der Raumtemperatur im Rahmen einer computergesteuerten Gebäudeklimatisierung muß im Gegensatz zu einer nicht-kontextsensitiven, in Abb. 14(a) auf der gegenüberliegenden Seite dargestellten Steuerungsanweisung bei deren kontextbezogener Variante die aktuelle Aktivität in den einzelnen Räumen berücksichtigt werden, so dass sich die in Abb. 14(b) dargestellte indirekte Adressierung der betreffenden Räume ergibt.*

Um diese Art von indirekter Adressierung möglichst effektiv gestalten zu können, werden zahlreiche, z. T. auch redundante Verknüpfungen zwischen den Kontextobjekten benötigt bzw. gewünscht. Es ergeben sich die folgenden Kriterien.

- Darstellung von Assoziationen zwischen Realweltobjektmodellen

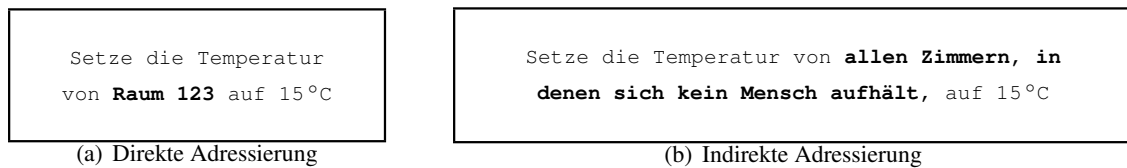


Abbildung 14: Adressierungsarten

- Möglichkeit zur Definition weiterer Assoziationen
- Möglichkeit zur flexiblen Gestaltung von Assoziationen, etwa hinsichtlich deren Abhängigkeit von anderen Informationen im Modell

2.4.3 Kontextbezogene Modellanforderungen

Die im folgenden dargestellten Anforderungen erwachsen aus den spezifischen Eigenschaften von Kontextinformationen und gelten daher speziell für deren Darstellung.

Anforderung 2 – *Berücksichtigung der Multidimensionalität der Welt*

Kontext ist kein eindimensionales, homogenes oder skalares Konzept, sondern umfasst prinzipiell sämtliche in einem Zusammenhang relevanten Informationen. Daher können verschiedene Kontextdimensionen, d. h. separate orthogonal kombinierbare Aspekte der Umgebung des Nutzers unterschieden werden (vgl. Kap. 2.1.1 („Begriffsbestimmung“, S. 13), Tab. 1 auf Seite 13). Ein Kontextmodell muß verschiedene Kontextdimensionen getrennt modellieren und flexibel miteinander kombinieren können. Die im folgenden dargestellten Anforderungen dienen zur Strukturierung dieses Anforderungskomplexes.

Anforderung 2a – *Separate Strukturierung von Datendimensionen*

Das Kontextmodell muß in der Lage sein, Kontextdimensionen separat zu modellieren und individuell zu strukturieren. Dazu muß es möglich sein, für jede Kontextdimension getrennte Ontologien zu verwenden. Dies kann durch die folgenden Kriterien ausgedrückt werden:

- Modellseitige Unterscheidung verschiedener Datendimensionen,
- Separate Ontologien für jede Datendimension.

Anforderung 2b – *Kombination separat strukturierter Datendimensionen*

Bei der Modellierung konkreter Realweltobjekte müssen die gemäß Anf. 2a separat modellierten Kontextdimensionen auf flexible Weise miteinander kombiniert werden können, d. h. Realweltobjektmodelle müssen flexibel in verschiedenen Kontextdimensionen eingeordnet werden und die dort definierten Attribute aufweisen. Es ergeben sich die folgenden zu erfüllenden Kriterien:

- Ableitung von Realweltobjektmodellen von verschiedenen Klassendefinitionen aus unterschiedlichen Ontologien
- Kombination der in verschiedenen Ontologien definierten Attribute

Anforderung 3 – *Berücksichtigung der Dynamik der Welt*

Der Kontext kann sich in verschiedener Hinsicht verändern. Zum einen kann der aktuelle Kontext binnen Sekunden wechseln, etwa wenn ein Fußgänger ein KFZ besteigt (die Handlungsmöglichkeiten und Anforderungen von Fußgängern und Autofahrern unterscheiden sich enorm, vgl. Kap. 6.2 („Funktionale Validierung“, S. 175)), zum anderen entwickelt sich die Welt permanent weiter, was auch von den diese Welt repräsentierenden Datenmodellen reflektiert werden muß. Dieser Anforderungsbereich wird durch die folgenden Anforderungen beschrieben

Anforderung 3a – *Dynamische Elemente und Dynamikdeskriptoren*

Konventionelle betriebliche Daten müssen i. d. R. einen hohen Grad an Datensicherheit und Konsistenz aufweisen, was sich auf die Gestaltung der diese Daten verarbeitenden betrieblichen Informationssysteme, insbesondere die dort verwendeten Konzepte zur Datenhaltung auswirkt, welche vor allem hinsichtlich Transaktionssicherheit und Durchsatz optimiert werden (vgl. [Gos⁺03, S. 3]).

Im Gegensatz dazu macht eine transaktionsorientierte Verarbeitung bei Kontextinformationen wenig Sinn, da diese eine durchwachsene Datenqualität aufweisen und damit nicht den strengen Konsistenzanforderungen wie konventionelle betriebliche Daten unterliegen. Ein wichtigeres Kriterium ist deren Aktualität, da die Qualität von Kontextinformationen im zeitlichen Verlauf nachläßt und auf veralteten bzw. unzureichenden Kontextdaten beruhende Adaptionen daher eher als belästigend denn nützlich empfunden werden. Deshalb sollte ihre Verarbeitung lieber in Richtung Echtzeitnähe als Transaktionssicherheit optimiert werden, denn eine transaktionale Verarbeitung bedingt zusätzliche zeitliche Verzögerungen bei konkurrierenden Aktualisierungsoperationen.

Die Gestaltung des Datenmodells sollte sich somit eher an den dynamischen Aspekten der modellierten Informationen orientieren. Zur Auswahl steht dabei zunächst die Möglichkeit, Inhalte aus den Originalquellen auszulesen und in der Datenbank vorzuhalten, wodurch diese sehr effizient an Kontextverbraucher ausgeliefert werden können. Andererseits können die bei sehr schnell veränderlichen Daten häufigen Aktualisierungen der Datenbank vermieden werden, wenn statt des Inhalts nur eine Referenz auf die Originaldaten in der externen Kontextquelle vorgehalten und diese bei Bedarf direkt zugegriffen wird. Zwar sind diese Dinge implementationsabhängig, doch sollte das Datenmodell entsprechende Entscheidungen durch eine Beschreibung der Dynamik, d. h. der Änderungshäufigkeit der betreffenden Datenfelder unterstützen, was durch die folgenden Kriterien ausgedrückt wird:

- Unterscheidung zwischen statischen und dynamischen Inhalten
- Darstellung der Änderungshäufigkeit einzelner Datenfelder

Ebenfalls diesem Anforderungskomplex zuzurechnen ist die Darstellung von *Aktualisierungsabhängigkeiten* zwischen einzelnen Kontextvariablen. Da diese Anforderung jedoch noch weiteren Anforderungsbereichen zuzuordnen ist, wird sie erst in Anf. 6i („Integration heterogener Kontextinformationen“, S. 40) diskutiert.

Anforderung 3b – *Generalisierende, erweiterbare Ontologien*

Nicht nur zur Strukturierung einer großen Zahl von Konzepten haben sich hierarchische Klassenstrukturen und das damit verbundene Prinzip der Vererbung aufgrund ihrer redundanzarmen Darstellungsweise bewährt (vgl. [Atk⁺95, S. 7 f]), auch die Erweiterung von Ontologien um neuartige Konzepte oder Kontextquellen, die in der sich permanent weiterentwickelnden Realwelt von Zeit zu Zeit auftauchen und auch von existierenden Anwendungen zu verarbeitenden sind, kann mittels einer hierarchischen Ontologie relativ ausdrucksstark gestaltet werden (vgl. [Lei⁺02, S. 3]). Um diese neuartigen Konzepte wenigstens ansatzweise im Begriffsmodell einer (nicht veränderbaren) existierenden Anwendung zu beschreiben und diese Anwendung daher in die Lage zu versetzen, die neuen Konzepte wenigstens ansatzweise interpretieren und verwenden zu können, müssen die neuen Konzepte als Subklassen bereits existierender, einer Anwendung bekannter Konzepte modelliert werden.

Dabei ist zu beachten, dass bestehende Ontologien strukturell nicht *verändert*, sondern nur *erweitert* werden sollten, um deren Kompatibilität hinsichtlich bestehender Anwendungen nicht einzuschränken. Um die verschiedenen Versionen einer Ontologie zu unterscheiden, sollen neuere Versionen unter einer höheren Versionsnummer oder einem anderen Namen dargestellt werden. Strukturell *veränderte* (d. h. nicht nur erweiterte) Ontologien müssen hingegen zwingend unter einem neuen Namen dargestellt werden. Eine Methodik zur Darstellung kontextbezogener Informationen sollte also die folgenden Kriterien erfüllen:

- Definition von Realweltobjektrepräsentationen mittels hierarchischer Ontologien
- Definition von Assoziationen mittels hierarchischer Ontologien
- Sichtbarkeit der Klassenhierarchie auf Seiten der Clientanwendungen, um die dadurch repräsentierten ontologischen Informationen zur Einordnung von Realweltobjekttypen nutzen zu können

Anforderung 3c – *Darstellung der Historie von Kontextvariablen*

Historie stellt eine zusätzliche, zu allen anderen Kontextdimensionen orthogonale Dimension von Kontextdaten dar, deren Analyse nicht nur die von LAMMING beschriebenen kontextbezogenen Retrieval-funktionalitäten in kontextsensitiv deskribierten Sitzungsprotokollen ermöglicht (vgl. Kap. 3.4.1.3 („Information Retrieval“, S. 59)), sondern auch Entwicklungstrends offenbart, die wiederum zur Extrapolation zukünftiger Kontextsituationen verwendet werden können (vgl. [Sch⁺99b, S. 894]). Dazu wird i. d. R. versucht, typische Situationen in den historischen Kontextdaten zu identifizieren und basierend auf dem Wissen über den Ablauf der Dinge mögliche zukünftige Szenarien zu projizieren, wozu neben Data Mining-Technologien komplexe Mustererkennungsverfahren verwendet werden (vgl. [Har⁺03]). Einige Anwendungen, welche die Erkennung von Entwicklungstrends historischer Kontextinformationen zur Assistenz von Nutzern verwenden, werden von HARVEL ET AL. beschrieben (vgl. [Har⁺03, S. 2 ff]). Daher stellt die Darstellung der Historie von Kontextinformationen eine wichtige Komponente eines Weltmodelles dar und sollte die im folgenden dargestellten Kriterien erfüllen.

- Repräsentation vergangener Inhalte von Kontextvariablen
- Kenntnis des Zeitpunktes, an dem dieser Wert gültig war
- Einfacher und intuitiver Zugriff auf alle historischen Informationen einer Kontextvariablen

Anforderung 4 – *Darstellung der Datenqualität von Kontextvariablen*

Name	Beschreibung
Aktualität	Kontextdaten veralten sehr schnell. So stellt die Ortsinformation eines beweglichen Objektes auch bei Kenntnis der damaligen Geschwindigkeit mit zunehmendem zeitlichen Abstand der Messung eine zunehmend unsichere Spekulation über den aktuellen Ort des Objektes dar (vgl. [ScGe01]).
Verfügbarkeit	Nicht jedes Kontextdatum kann zu jedem beliebigen Zeitpunkt erfasst werden. Manche Aspekte von Realweltobjekten werden nur in speziellen Situationen sichtbar und sind vor diesen Situation nicht verfügbar.
Vollständigkeit	bestimmte Kontextinformationen wurden nicht erfasst und fehlen im Weltmodell.
Widersprüchlichkeit	Teile der Daten widersprechen sich oder es sind mehrere alternative, sich widersprechende Aussagen über ein Realweltobjekt vorhanden.
Genauigkeit, Granularität	Die Daten liegen in einer zu groben Auflösung vor, da die kleinste vom Sensor erkannte Maßeinheit zu grob für die gegebene Problemstellung ist.
Präzision	Variation des Messwertes bei direkt hintereinander durchgeführten gleichartigen Messungen.
Korrektheit	Unterschied zwischen gemessenem und tatsächlichen Wert, d. h. die Daten spiegeln den Status der realen Welt nicht wider. Dies hängt von der <i>Zuverlässigkeit</i> des verwendeten Sensors ab. So kann eine Gesichtserkennung falsche Ergebnisse liefern. Auch die Informationen aus dem Terminplan des Nutzers sollten als eher unverbindlich aufgefasst werden.
Vertrauenswürdigkeit	die Daten stammen aus einer Kontextquelle, der nur eingeschränkt vertraut werden kann, weil sie z. B. nicht zertifiziert ist.

Tabelle 5: Qualitätsmängel von Kontextdaten

Allgemein in der Community anerkannt ist die Tatsache, dass Kontextdaten prinzipbedingt verrauscht und ungenau sind (vgl. [Hul⁺97, S. 8] [Hen⁺02, Dey⁺00a, Ebl⁺01, GrSa01, Lei⁺02]). Im Gegensatz zu klassischerweise von Datenbanken verwalteten Informationen, die von menschlichen Nutzern eingegeben werden und vom Datenbanksystem abgesehen von offensichtlichen Konsistenzfragen nicht in Frage zu stellen oder eigenmächtig zu verwerfen sind, stellen Kontextdaten nur *Vermutungen* eines Computersystems hinsichtlich der Realität dar und unterliegen deshalb dem selben Grad an Subjektivität wie die Weltansicht menschlicher Nutzer, d. h. sie können widersprüchlich, unvollständig, unsicher und ungenau sein. Im Einzelnen können die in Tab. 5 dargestellten Qualitätsmängel bei Kontextdaten auftreten (vgl. [Hen⁺02, S. 177 ff], [Hong01]). Daher ist es notwendig, im Rahmen von Entscheidungsprozessen die durchaus variable Qualität der dabei verwendeten Kontextinformationen zu kennen.

Zur Überwindung dieser durch Subjektivität und unvollständige Weltansicht verursachten Probleme müssen die selben Verfahren eingesetzt werden, die auch von Menschen bei Unsicherheit über die Realität verwendet werden. Dazu gehört die Kombination mehrerer, möglichst verschiedener, subjektiver Sichtweisen und darauf folgend die Auswahl der wahrscheinlichsten Alternative aus einer Menge von Aussagen basierend auf den Eigenschaften der jeweiligen Quelle. Diese Mechanismen werden im Rahmen der Integration von Kontextdaten überhaupt erst möglich und können zur Reduzierung der dargestellten Probleme beitragen. Auch sollten Kontextinformationen immer als *optionale* Parameter ver-

Kontextvariable	Qualitätsattribut	Metrik
Ort	Genauigkeit	Standardabweichung des Lokalisierungs-sensors in Millimetern
Zeit	Genauigkeit	Standardabweichung der Uhr in Millisekunden
Ort	Aktualität	Zeitstempel des Meßvorganges
Nutzeraktivität	Sicherheit der Quelle über die Korrektheit der Information	Gewissheit der zugrunde liegenden Informationen als Wahrscheinlichkeitsmaß
Nutzeraktivität	Genauigkeit der Quelle	Wahrscheinlichkeitsmaß

Tabelle 6: Qualitätsattribute für Kontextvariablen (vgl. [Hen⁺02, S. 178])

standen werden, d. h. es sollte immer eine standardmäßige Vorgehensweise für den Fall unzureichender Kontextinformationen bekannt sein. Im einzelnen sind die folgenden Kriterien zu erfüllen:

Anforderung 4a – *Bereitstellung von Deskriptoren zur Beschreibung der Datenqualität*

Die wichtigsten Aspekte der Datenqualität von Objektattributen sollten durch entsprechende *QoI*-Attribute (Quality of Information, vgl. [Lei⁺02, S. 2 f]) beschrieben werden können, um diese Aspekte bei der Verwendung dieser Informationen berücksichtigen zu können. Näheres dazu findet sich in den Arbeiten von WANG ([Wan⁺95]).

- Darstellung von Qualitätsdeskriptoren zu Attributwerten
- dies muß für alle auftretenden Attributwerte möglich sein

Anforderung 4b – *Datentypspezifische Qualitätsdeskriptoren*

Unterschiedliche Kontextvariablen haben u. U. unterschiedliche Qualitätsaspekte, -maßstäbe und -metriken. Tabelle 6 gibt ein paar Beispiele für mögliche Qualitätsattribute und -metriken von Kontextvariablen. Diese flexiblen Beschreibungsmethoden, welche unterschiedliche, auf den jeweiligen Kontextdatentyp spezialisierte Qualitätsattributtypen vorsehen, müssen im Kontextmodell vorgesehen sein, was durch die folgenden Kriterien beschrieben werden kann:

- Unterscheidung verschiedener Qualitätsattribute pro Kontextvariable
- Auswahl der Qualitätsattributtypen in Abhängigkeit vom Typ der Kontextvariable
- Auswahl der Qualitätsmetriken der Qualitätsattribute in Abhängigkeit vom Typ der Kontextvariable

2.4.4 Datensicherheit

Gesellschaftlich umstritten ist beim Einsatz kontextbezogener Technologien vor allem die zunehmende Überwachung der Privatsphäre durch elektronische Systeme und die sich daraus ergebenden sozialen und gesellschaftlichen Veränderungen (vgl. [Sloa92, Coy92, AgRo97]). Die Menschen vertrauen dabei prinzipiell keinen Geräten, die sie nicht verstehen bzw. kontrollieren können (vgl. [Harp95], [Hul⁺97,

S. 10 ff]). Trotz der in diesem Bereich ebenfalls häufig anzutreffenden Gleichgültigkeit kann die Ablehnung von überwachender bzw. fälschlicherweise als solche interpretierten Technologie derart grundlegend sein, dass der Schutz der Privatsphäre der überwachten Nutzer letztlich das entscheidendste K.O.-Kriterium für kontextsensitive Lösungen darstellt und deshalb als eine der wichtigsten Anforderung an kontextbezogene Anwendungen angesehen werden muß (vgl. [Coy92, Hong04], [Lei⁺02, S. 2]). Während im folgenden diesbezügliche methodische Anforderungen an Verfahren zur Darstellung privater Daten evaluiert werden, finden sich weitergehende Überlegungen zu diesem Aspekt im Rahmen der Problembetrachtung von *Context Maps* in Kap. 6.4.2 („Schutz der Privatsphäre“, S. 199).

Anforderung 5 – *Schutz der Privatsphäre der überwachten Nutzer*

Während die anderen in dieser Arbeit dargestellten Anforderungen *optimistischer* Natur sind, d. h. es anderen Instanzen möglichst leicht machen sollen, nützliche Funktionalitäten für den Nutzer zu erbringen (vgl. [Pove99, GrHo03]), werden an dieser Stelle *pessimistische* Anforderungen, d. h. Verfahren, um es anderen Instanzen möglichst schwer machen sollen, die Daten des Nutzers zu mißbrauchen, diskutiert (vgl. [Hong04]). Letztlich gilt es, anwendungs- und nutzerabhängig einen Kompromiss zwischen beiden sich nur bedingt widersprechenden, sondern teilweise auch orthogonalen Anforderungsbereichen angemessen Rechnung zu tragen.

Anforderung 5a – *Datenschutzdeskriptoren für einzelne Daten*

Viele Nutzer fühlen sich wesentlich sicherer und legen daher Wert darauf, die zwingend einzuhaltenen Verfahrensanweisungen für ihre persönlichen Daten selbst festlegen zu können und darüber auch Feedback zu erhalten (vgl. [BaDe03a, BeSe93]). Dazu zählen neben Zugriffsrechten z. B. auch Hinweise, ob die betreffenden Daten mit anderen Informationen kombiniert werden dürfen. Notwendig sind daher in jedem Fall detaillierte Informationen über den Eigentümer und Verwendungsbeschränkungen privater Daten, welche folgenden Kriterien genügen:

- Darstellung der Eigentumsverhältnisse von Informationen
- Modellierung von unterschiedlichen Zugriffsrechten auf einzelne Daten
- Modellierung von Verfahrensanweisungen für die Daten

Dabei muß durch das Datenbanksystem sichergestellt werden, dass diese Zugriffsrechte zwingend gelten und vom Nutzer verändert werden können.

Anforderung 5b – *Darstellung von Akteuren*

Um die Zugriffsrechte im praktischen Alltag angemessen anwenden zu können, ist es notwendig, die Identität der beteiligten Akteure, d. h. der Besitzer von Daten, der anfragenden Instanzen und der diese Instanzen steuernden Anwender korrekt und innerhalb des verwendeten Sicherheitsmodells eindeutig festzustellen. Da diese Informationen im Prinzip auch Kontextinformationen darstellen, werden sie an dieser Stelle in den Anforderungskatalog für Kontextmodelle aufgenommen, obwohl sie auch dem davon unabhängigen Sicherheitsmodell zuzurechnen sind. Dies zeigt, dass sich Kontextinformationen nicht nur negativ auf den Schutz der Privatsphäre der Nutzer auswirken müssen, sondern durch die Unterstützung

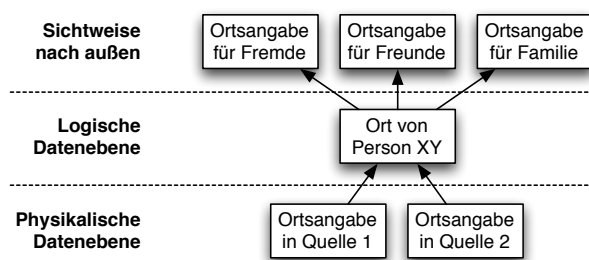


Abbildung 15: Variable Repräsentationen einer Kontextvariablen für verschiedene Zielgruppen (vgl. [Hon⁺03, S. 52])

technischer Sicherheitsmaßnahmen auch zu deren Verbreitung und einfacheren Nutzung und damit zu einer auch für Normalanwender höheren Datensicherheit führen können. Notwendig sind also die durch die folgenden Kriterien beschriebenen Identifikationsmechanismen.

- Darstellung von Eigentümern der Informationen
- Darstellung von anfragenden Anwendungen
- Darstellung von anfragenden Anwendern

Anforderung 5c – Unterscheidung von verschiedenen Vertrauensstufen

Erst wenn die beschriebenen Datenschutzmechanismen funktionieren, sind Endanwender durchaus bereit, ausgewählte Teile der Kontrolle über ihre privaten Daten schrittweise auch externen Instanzen zu überlassen, wenn der dafür erhaltene Nutzen hoch und das zu erwartende Mißbrauchspotenzial gering genug ausfallen (vgl. [BaDe03b]).

Da die Vertrauenswürdigkeit von externen Instanzen dabei durchaus unterschiedlich ist, sollen die an verschiedene Instanzen übermittelten Informationen eine u. U. unterschiedliche Genauigkeit bzw. Detaillierung haben. Daher müssen die nutzerbezogenen Daten in verschiedenen Genauigkeits- und Detaillierungsstufen verfügbar sein. Als Beispiel seien an dieser Stelle die in Abb. 15 dargestellten unterschiedlichen Genauigkeiten bei der Angabe des aktuellen Ortes genannt. Insgesamt sind dabei die folgenden Kriterien relevant:

- Darstellung von verschiedenen Vertrauensstufen
- Zuordnung von Vertrauensstufen zu externen Instanzen
- Bereitstellung von verschiedenen präzisen Repräsentationen ein und desselben Datums
- Zuordnung von Repräsentationen zu Vertrauensstufen

2.4.5 Integration heterogener Kontextinformationen

Die sowohl strukturelle als auch inhaltliche Integration heterogener, d. h. aus unterschiedlichen Quellen stammender Kontextinformationen stellt das zentrale Element dieser Arbeit dar und wird im folgenden genauer analysiert.

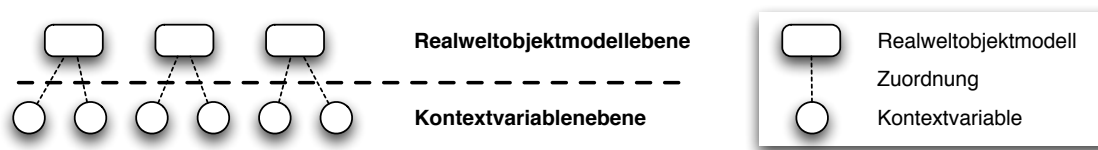


Abbildung 16: Realweltobjektbezogene Strukturierung von Kontextvariablen

Anforderung 6 – *Semantische Integration von Kontextinformationen aus verschiedenen Quellen in inhaltlicher und struktureller Hinsicht*

Dieser Anforderungskomplex wird direkt im Rahmen der Validierung von These 3 („Möglichkeit der Integration heterogener Kontextinformationen“) verwendet. Für die semantische Integration von heterogenen kontextbezogenen Informationen ergeben sich die im folgenden beschriebenen Anforderungen.

Anforderung 6a – *Realweltobjektbezogene Strukturierung des Kontextmodells*

Ein brauchbares Realweltmodell sollte nicht nur die bekannten Kontextvariablen auflisten. Um sich gezielt Informationen über bestimmte Realweltobjekte aus dem Kontextmodell besorgen zu können, sollte dieses zunächst sämtliche Realweltobjekte repräsentieren und dieses Realweltmodell zur Adressierung der entsprechenden Kontextvariablen verwenden (vgl. Abb. 16). Da sie zur Navigation und Interpretation anderer Kontextobjekte notwendig sein können, müssen dabei auch diejenigen Realweltobjekte, über die keine weiteren Informationen bekannt sind, im Modell dargestellt werden. Dies führt zu folgenden Anforderungskriterien:

- Repräsentation aller bekannten Realweltobjekte unabhängig von den über sie bekannten Kontextinformationen
- Zuordnung von Kontextvariablen zu Realweltobjektrepräsentationen

Anforderung 6b – *Eineindeutige Repräsentation von Realweltobjekten*

Um verschiedene Kontextvariablen eineindeutig ihren jeweiligen Realweltobjektmodellen zuordnen zu können, müssen die betreffenden Realweltobjektmodelle innerhalb des Kontextmodells eineindeutig identifiziert werden, d. h. jedes Realweltobjekt darf durch nur ein Realweltobjektmodell repräsentiert und jedes Realweltobjektmodell darf nur ein Realweltobjekt beschreiben. Als Schlüssel zur Adressierung von Realweltobjektmodellen muß daher ein eindeutiges Merkmal des Realweltobjektes verwendet werden und im Realweltobjektmodell muß ein eindeutiger Zeiger auf das entsprechende Realweltobjekt vorhanden sein. Die eineindeutige Zuordnung von Realweltobjekt und Modell ist auch im Sinne des in Kap. 2.1.1 („Begriffsbestimmung“, S. 14) definierten Semantikbegriffes notwendig. Zusammengefasst ergeben sich die folgenden Kriterien:

- Anwendungsübergreifende, globale Adressierung von Realweltobjekten und ihren Modellen
- eindeutige Zuordnung von Realweltobjektmodellen zu Realweltobjekten
- eindeutiger Link auf das betreffende Realweltobjekt im Realweltobjektmodell

Anforderung 6c – *Explizite Ontologien*

Die Realität ist beliebig komplex und kann daher weder von Anwendungen oder Anwendern vollständig abgebildet noch in ihrer Gänze verstanden und interpretiert werden. Dies ist bekanntermaßen nur mit *Modellen*, d. h. vereinfachten Abbildern der Realität möglich. Daten und Informationen, welche die Realwelt modellieren, stellen also *subjektive Projektionen* der Wirklichkeit dar, wobei unter *Projektion* in diesem Zusammenhang eine Abbildung der Realität und unter *Subjektivität* die Individualität der bei dieser Projektion verwendeten Perspektive und Zielstellung verstanden wird. Bereits existierende Daten- und Informationsbestände stellen eine solche subjektive, gemäß einer bestimmten Weltsicht und Ontologie formulierten Sichtweise auf die Realwelt dar. Der Vorgang der *Subjektivierung*, d. h. der Formulierung der Informationsmenge in einer eigenen Perspektive bzw. Ontologie, wird dabei i. d. R. implizit während der Erstellung der jeweiligen Modelle vollzogen und in Form des verwendeten Datenschemas bzw. der verwendeten Ontologie dokumentiert.

Dabei haben die als Kontextquellen verwendeten betrieblichen Datenbestände eine i. d. R. *wirtschaftlich* verwendungsorientierte Motivation und Perspektive, die nicht zwangsläufig mit der in dieser Arbeit benötigten *kontextbezogenen* Perspektive übereinstimmt. Die Wiederverwendung von betrieblichen Datenbeständen in einem kontextbezogenen Zusammenhang erfordert daher die Uminterpretation der in betrieblichen Informationsbeständen existierenden Informationen in eine kontextbezogene Perspektive.

Diese Transformation sollte durch geeignete Werkzeuge weitgehend automatisiert erfolgen, was jedoch an dieser Stelle mit Hinweis auf den Ausblick in Kap. 7.2 auf S. 208 nicht weiter verfolgt werden soll. Damit dies möglich ist, müssen beide Ontologien, d. h. die Ausgangs- und die Zielontologie explizit formuliert sowie diverse interontologische Informationen bekannt sein, wobei letztere ebenfalls nicht Bestandteil dieser Arbeit sein sollen.

Auch erfordert die kollektive Nutzung der integrierten Kontextmodelle durch mehrere Anwendungen eine explizite Darstellung der dabei zugrundeliegenden Ontologien als Basis eines Konsenses hinsichtlich der Interpretation der dargestellten Informationen durch die verschiedene Anwendungen. Zusammengefasst ergeben sie die folgenden Kriterien:

- Nutzung von Ontologien zur Definition der verwendeten Begriffe und Konzepte
- Explizite Darstellung der Ontologie
- Unterstützung mehrerer Ontologien

Anforderung 6d – *Darstellung von externen Kontextquellen*

Dieser Arbeit liegt das in Kap. 2.3 („Verteilungsmodell für Kontextinformationen“, S. 24) beschriebene Verteilungsmodell für Kontextinformationen und damit die Vision zahlreicher externer Kontextquellen zugrunde, deren Inhalte in ein quellenübergreifendes Realweltmodell zu integrieren sind. Um die Kontextvariablen aus den externen Kontextquellen modellseitig möglichst flexibel in das Realweltmodell zu integrieren, ist die Darstellung der externen Kontextquellen innerhalb des Modells notwendig. Dies ermöglicht die vollautomatische Konfiguration der zum Zugriff auf die einzelnen Kontextquellen anzuwendenden Zugriffsverfahren. Diese Anforderung kann in den folgenden Kriterien formuliert werden:

- Darstellung aller zu verwendenden Kontextquellen,

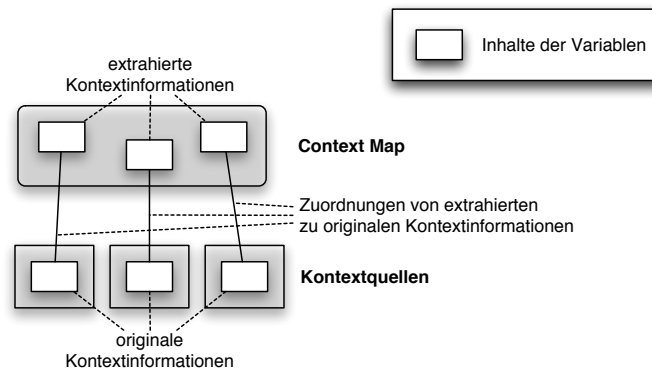


Abbildung 17: Doppelte Datenhaltung

- Angabe aller zum Zugriff auf die Kontextquelle benötigten Informationen, d. h. Quellentyp, Zugriffsverfahren, Adresse, Logindaten,

Anforderung 6e – Darstellung der Inhalte der einzelnen Kontextvariablen im Kontextmodell

Um das Verteilungsproblem zu lösen, d. h. als einziger Zugriffspunkt für eine verteilte Kontextinformationsbasis zu fungieren, müssen die Inhalte der betreffenden Kontextvariablen im Modell selbst dargestellt werden. Dies ist auch für deren Verarbeitung und Kombination, etwa die Vereinigung synonyme Inhalte aus verschiedenen Kontextquellen, notwendig, welche im Anschluß separat betrachtet wird. Dabei müssen die Inhalte im integrierten Kontextmodell bei Veränderungen in den Kontextquellen zeitnah aktualisiert werden. Insgesamt ergeben sich die folgenden Kriterien.

- Darstellung des Inhaltes referenzierter externer Informationen im Kontextmodell

Anforderung 6f – Verbindung von Inhalten im Kontextmodell zu den Originaldaten in den externen Quellen

Um die verschiedenen Kontextinformationen verarbeiten und miteinander kombinieren zu können, müssen sie innerhalb des Modells nicht nur referenziert, sondern auch inhaltlich dargestellt werden. Dabei muß die Verbindung zu den originalen Daten innerhalb der einzelnen Kontextquellen beibehalten werden, um dortige Veränderungen zu erkennen und automatisiert in das Kontextmodell übertragen zu können (vgl. Abb. 17). Daher ergibt sich in der Praxis zwar eine redundante Datenhaltung, wobei die Daten in der *Context Map* jedoch transparent mit den Originaldaten synchronisiert werden und damit in der Praxis keine Inkonsistenzen auftreten können.

Die Adressierung der Datensätze in den Kontextquellen kann z. B. durch eine Referenz in Form einer URI auf den betreffenden Datensatz in der externen Kontextquelle erfolgen (vgl. [Ber⁺98]). Die folgenden Kriterien beschreiben diese Anforderung näher.

- Möglichkeit zur Zuordnung von Datensätzen im Kontextmodell zu Datensätzen in externen Kontextquellen
- Einbeziehung von Kontextquellenmodellen zur redundanzfreien Darstellung von Zugriffsinformationen auf die entsprechenden Kontextquellen

- Möglichkeiten zur flexiblen Adressierung der Informationen innerhalb der Kontextquellen

Anforderung 6g – *Fusion synonymer Kontextvariablen*

Zur Ableitung integrierterer, höherwertigerer Ansichten von Kontext innerhalb *eines* Kontextmodells sind lediglich die beiden Operationen zur Aggregation und Interpretation notwendig, wobei unter *Aggregation* das Zusammenführen von elementaren Kontextvariablen zu Realweltobjektrepräsentationen und unter *Interpretation* die Ableitung höherwertigerer Aussagen aus elementaren Informationen verstanden wird (vgl. [Dey00b, S. 40 f]).

Bei der Integration *mehrerer* Realweltmodelle treten jedoch zwangsläufig synonyme, d. h. den selben Realweltaspekt des selben Realweltobjektes beschreibenden Kontextvariablen auf. Diese müssen durch eine weitere, *Fusion* genannte Operation in angemessener Weise vereinigt, d. h. inhaltlich kombiniert werden, wobei bei widersprüchlichen, d. h. unvereinbaren Kontextvariablen statt einer inhaltlichen Kombination eine Auswahl der wahrscheinlichsten Aussage aus den alternativen Kontextvariablen durchgeführt wird. Im Gegensatz zur *Interpretation* verändert die *Fusion* den Abstraktionsgrad der betreffenden Variablen nicht. Es ergeben sich die folgenden Kriterien, die ein Kontextmodell zur Unterstützung einer Fusionsoperation erfüllen muß:

- Darstellung aller synonymen, d. h. für ein und dasselbe Realweltobjektattribut kandidierenden Kontextvariablen
- Darstellung des zur Fusion anzuwendenden Verfahrens

Anforderung 6h – *Integration von Methodik in das Datenmodell*

Zwischen einzelnen kontextbezogenen Anwendungen können nicht nur abstrakte Begrifflichkeiten (Ontologien) und konkrete Weltansichten (die eigentlichen Kontextinformationen), sondern auch damit zusammenhängende Transformations- und Interpretationsroutinen, etwa Verfahren zur Interpretation von Ortskoordinaten oder zur Gesichtserkennung geteilt werden. Diese informationsbezogenen Methodiken können weniger einzelnen Anwendungen, sondern eher den entsprechenden Informationsstrukturen zugerechnet werden und sollten deshalb zusammen mit diesen dargestellt werden, wobei der Schwerpunkt der entstehenden Informationsstruktur im Gegensatz zur objektorientierten Programmierung jedoch eindeutig auf der Seite der Daten liegt (vgl. [Atk⁺95, S. 5 f]).

Die Darstellung von kontextbezogener Methodik im Kontextmodell kann in begrenztem Maße auch zu einer Trennung von Geschäfts- und Kontextlogik führen, was aufgrund der folgenden Gründe wünschenswert ist:

- die Modellierung dieser zueinander orthogonalen Aspekte sollte durch separate Personen erfolgen. Anwendungen werden i. d. R. von informatisch trainierten Programmierern erstellt, während Kontextmodelle in den Zuständigkeitsbereich von den jeweiligen z. T. betriebswirtschaftlich orientierten Domänenexperten fallen.
- Die zusätzliche Verarbeitung von Kontextinformationen erhöht die Komplexität des Anwendungscodes

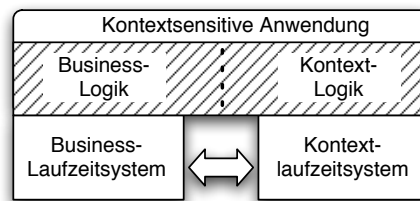


Abbildung 18: Trennung einer kontextsensitiven Anwendung in einen Business- und Kontextteil

- Die häufigen Änderungen an der Kontextlogik würden jedes Mal das erneute Kompilieren und Installieren des gesamten Anwendungscode erfordern,
- Fehler im kontextverarbeitenden Teil von Anwendungen könnten sich auf die Konsistenz betrieblicher Anwendungsdaten auswirken,
- Kontextbezogene Funktionen können genau wie kontextbezogene Informationen durch mehrere Anwendungen verwendet werden

Kontextbezogene Anwendungen sollten entsprechend in einen *Business*- und einen *Kontextteil* getrennt werden, wobei der Businesssteil anwendungsbezogene und der Kontextteil kontextbezogene Code- und Datenelemente enthält und beide Funktionalitäten durch jeweils entsprechend spezialisierte Komponenten und Laufzeitsysteme realisiert werden (vgl. Abb. 18). Anwendungslogik wird im betrieblichen Umfeld vor allem durch Workflowsysteme und EJB-Container, kontextbezogene Funktionalität durch entsprechende Kontextdatenbanken oder wie von HONG beschrieben ganze Kontextinfrastrukturen, die einen komplett anderen Aufbau und eine andere Arbeitsweise als etwa Workflowsysteme aufweisen, erbracht (vgl. [Hong01, S. 3]).

Darüber hinaus ermöglicht die Einbettung von Codeblöcken in Datenstrukturen einen gewissen Grad von *Selbstorganisation*, indem sich die Informationsbasis quasi selbstständig fehlende Informationen etwa durch Auswertung bereits bekannter Informationen besorgt oder sich entsprechend den bestehenden Kontextverhältnissen selbsttätig umorganisiert. Schließlich ist eine Kontextdatenbasis, in die nicht nur die benötigten Informationen und Metainformationen, sondern auch die dazugehörige Methodik eingebettet ist, auch in einem wesentlich umfassenderen Sinn replikationsfähig (vgl. Anf. 1c („Verteilte Darstellung und Speicherung“)). Es ergeben sich die folgenden Kriterien.

- Möglichkeit zur Darstellung von Operationen auf Inhalten innerhalb des Datenmodells
- Möglichkeit zum Aufruf externer Funktionen
- möglichst direkte Darstellung des Codes an der jeweils verwendeten Stelle in der Datenbasis
- Darstellung der Umstände, wann und wo der Code aufzurufen ist
- Möglichkeit zur Adressierung sämtlicher benötigter Parameter durch das Kontextmodell
- Einbeziehung beliebig vieler Kontextvariablen in Operationen
- Möglichkeit, die neu berechneten Inhalte in die Datenbasis einzufügen

Anforderung 6i – Darstellung des Datenpfades von Kontextinformationen

Um die aufgrund des dynamischen Charakters von Kontextinformationen häufig auftreten Aktualisierungen der Kontextinformationsbasis möglichst automatisch durchführen zu können und auf die auch wirklich betroffenen Teile des Kontextmodells zu begrenzen sowie die Qualität und Geheimhaltungsstufe von Informationen aufgrund deren Herkunft abschätzen zu können, ist die explizite Abbildung des *Datenpfades*, d. h. der konkreten Verarbeitungsschritte von *low level*- zu *high level*-Informationen notwendig (vgl. Anf. 3 („Dynamik“), Anf. 5c („Vertrauensstufen“)).

Dies beinhaltet die Darstellung von *Aktualisierungsabhängigkeiten*, d. h. welche Informationen mittels welcher Operationen aus welchen anderen Informationen berechnet werden (vgl. [Hen⁺02, S. 176 f]). Ohne diesbezügliche Vorkehrungen können durch veraltete, nicht aktualisierte Kontextinformationen falsche Entscheidungen durch kontextsensitive Systeme getroffen werden (vgl. [Efs⁺01]). Dabei gelten die folgenden Anforderungen:

- Darstellung von Aktualisierungsabhängigkeiten zwischen Kontextvariablen
- Darstellung der zur Neuberechnung der Datenbasis notwendigen Verfahren
- Darstellung, welche Kontextinformationen welche Rolle in welchen Operationen spielen

Anforderung 6j – *Einschränkung der Sichtbarkeit von Konzepten*

Praxisnahe Szenarien können eine enorme Vielzahl von Realweltobjektmodellen enthalten, wodurch die Realweltmodelle sehr unübersichtlich werden können, was unter dem Stichwort *Informationsüberflutung* auch in vielen anderen Bereichen des Informationsmanagements von Bedeutung ist. Notwendig sind daher Mechanismen, um die im aktuellen Zusammenhang irrelevanten Informationen herauszufiltern, d. h. die Darstellung der Realwelt hinsichtlich der gegebenen Aufgabenstellung in vereinfachender Weise zu adaptieren.

Möglich wird dies, da Kontextinformationen i. d. R. eine deutliche Lokalität aufweisen, d. h. sie sind nur in den seltensten Fällen global, sondern meist lediglich hinsichtlich bestimmter Orte, Zeitpunkte oder Problemstellungen, beispielsweise in der Umgebung ihres räumlichen oder zeitlichen Ursprunges von Interesse und verlieren mit zunehmendem Abstand davon deutlich an Relevanz (vgl. [ScGe01, S. 2 f]). In den meisten Fällen ist also zumindest die *Darstellung* sämtlicher bekannter Realweltobjektmodelle überhaupt nicht notwendig, da viele Realweltobjekte keine Relevanz hinsichtlich der gegebenen Aufgabenstellung besitzen. Um diese problembezogene Sichtbarkeitsbegrenzung durchführen zu können, muß der *Bezug* der Realweltobjekte im Kontextmodell dargestellt sein, d. h. zumindest das grobe Themengebiet, auf das sich die betreffenden Informationen sowie die aktuelle Aufgabenstellung des betreffenden Kontextverbrauchers beziehen.

Beispiel 4 *Der französische Version des Namens einer Stadt ist für einen englischen Touristen nicht relevant und sollte demzufolge nicht dargestellt werden. Ebenso sind für einen Touristen auf Besichtigungstour nur touristische Informationen relevant und nicht etwa Informationen über bestimmte Tagungshotels in der Nähe.*

HULL verweist auf die mit steigender Zahl von dargestellten Konzepten zunehmenden Skalierungsprobleme bei Verwendung eines globalen Namensraumes für Realweltobjekte, d. h. Probleme, eindeutige

Namen für sehr viele global sichtbare Konzepte zu finden (vgl. [Hul⁺97, S. 11]). Da die Menge an sinnvollerweise verwendbaren Begriffen i. d. R. geringer als die Anzahl der zu modellierenden Realweltkonzepte ist, treten irgendwann *Homonyme*, d. h. verschiedene Konzepte mit dem selben Namen auf, welche zwar verwendet werden können, dann aber getrennten Gültigkeitsbereichen zuzuordnen sind. Notwendig ist also die Unterteilung des globalen Namensraumes in logisch voneinander abgegrenzte Bereiche, wobei einzelne Konzepte zunächst nur innerhalb ihres Bereiches und erst bei tatsächlichem Bedarf global sichtbar sein sollen. Diese logischen Bereiche innerhalb des Kontextmodells sollten flexibel und realweltorientiert definiert und abgegrenzt werden können.

Beispiel 5 *Wenn Jeff Malone von „Klaus“ ohne weitere Angaben redet, kann es sich nur um seinen Bekannten Klaus Meier handeln, während John Doe, der Klaus Meier überhaupt nicht kennt, mit „Klaus“ seinen Bekannten Klaus Gerhard meint.*

Logische Bereiche im Kontextmodell würden in Beispiel 5 den beiden Personen *Jeff Malone* und *John Doe* zugeordnet werden, wobei diesen Bereichen dann die persönlichen Informationen der betreffenden Personen sowie die jeweils personenbezogenen Namen der ebenfalls genannten dritten Personen zugeordnet werden würden (vgl. nächste Anforderung). Die Anforderung kann schließlich in den folgenden Kriterien zusammengefasst werden.

- Möglichkeit zur Definition logischer Bereiche innerhalb der Datenbasis
- Möglichkeit, beliebig viele verschiedene Arten von logischen Bereichen parallel zu definieren
- Semantische Zuordnung dieser logischen Bereiche zu geeigneten Realweltkonzepten
- flexible Abgrenzung dieser logischen Bereiche
- Zuordnung von Inhalten oder bestimmten Teilen davon zu logischen Bereichen

Anforderung 6k – *Synonyme Benennung von Topics*

Fast alle Realweltobjekte und -konzepte werden in der Praxis synonym, d. h. mit mehreren Namen bezeichnet. Dies umfasst neben oft verschiedenen Bezeichnungen in Schrift- und Umgangssprache (beide stellen relevante Domänen für Kontextinformationen dar) auch unterschiedlich detaillierte, unterschiedlich formale sowie in verschiedenen natürlichen oder technischen Sprachen formulierte Bezeichnungen für ein und das selbe Realweltobjekt sowie weitere Spezialformen für Namen wie Spitznamen. Teilweise werden die synonymen Namen in verschiedenen, manchmal jedoch auch im gleichen Zusammenhang verwendet.

Beispiel 6 *Das Auto von Jeff Malone kann in den unterschiedlichen Situationen als „Ihr KFZ“, „der Wagen“, „die (verdammte) Karre“, „Ihr PKW“ oder „DD-KM 2000“ bezeichnet werden.*

Um ein Objekt trotz Verwendung synonyme Bezeichnungen sicher identifizieren zu können, müssen diese bekannt, d. h. im Realweltmodell dargestellt sein, was durch die folgenden Kriterien ausgedrückt wird.

- Angabe alternativ verwendbarer, d. h. synonyme Bezeichnungen für Realweltobjekte
- Eingrenzung der Sichtbarkeit einzelner synonyme Bezeichnungen auf spezielle Anwendungsbereiche (vgl. Anf. 6j („Einschränkung der Sichtbarkeit von Konzepten“))

2.4.6 Endnutzertauglichkeit

Die aus dem Kontextbezug resultierende enge Orientierung kontextsensitiver Systeme an der sich permanent weiterentwickelnden Realwelt führt zu häufigen Änderungen an den innerhalb dieser Systeme eingesetzten, die Realwelt abbildenden Kontextmodellen, welche i. d. R. durch Menschen durchgeführt werden. Da mathematische Konstrukte in der Lebenswelt der betrieblichen Anwender, die durch das Stereotyp „Praktiker“ beschrieben werden können, kaum eine Rolle spielen, werden zu formale Methoden in der betrieblichen Praxis i. d. R. abgelehnt, da sich die mathematischen Mittel zur Definition der formalen Sprachen nicht an der Lebenswelt, d. h. der Denkweise der betrieblichen Anwender orientieren und somit als Formalisierungsmittel wenig hilfreich sind, da sie nur teilweise verstanden und damit auch nicht in vollem Umfang bzw. sogar falsch eingesetzt werden können (vgl. [Mess99, S. 99 ff]). Daher stellen ergonomische, leicht erlern- und überblickbare sowie intuitiv wartbare Kontextmodelle auch im betrieblichen Umfeld einen relevanten Erfolgsfaktor für kontextbezogene Lösungen dar.

Neben betrieblichen Anwendungen zur Unterstützung von Entscheidungen und Tätigkeiten im betrieblichen Umfeld dienen kontextsensitive Anwendungen aber auch zur Unterstützung einer ganzen Palette von zwar auch kommerziellen, aber im Nutzeralltag anzufindenden Anwendungen. Die dabei zu berücksichtigenden individuellen Umstände des jeweiligen Nutzers müssen angemessen berücksichtigt werden, wobei es jedoch ökonomisch nicht sinnvoll bzw. technisch teilweise überhaupt nicht möglich ist, sämtliche potentiell auftretenden Situationen bereits zum Erstellungszeitpunkt einer kontextsensitiven Anwendung bzw. eines Kontextmodells vollständig und korrekt zu antizipieren (vgl. [Dey01a, S. 12]).

Besonders im privaten Bereich von Endnutzern sind bestimmte Teile der Realwelt letztlich nur ihm selbst bekannt und sollen auch keinen anderen Menschen bekannt werden, weshalb sie zwangsläufig auch von ihm selbst bereitgestellt werden müssen. Dies betrifft neben zu verwendenden Quellen privater Kontextinformationen z. B. auch die Modellierung von Beziehungen des Nutzers zu anderen Personen, wobei ggf. zwischen Freundschaft, Bekanntschaft und Kollegialität zu unterscheiden ist (vgl. [New⁺02]).

Da die betreffenden Informationen im Idealfall nur einmalig vom Nutzer abgefragt werden und dann sämtlichen anderen Anwendungen zur Verfügung stehen, wird dieser Mehraufwand von vielen Forschern als auch für Endverbraucher zumutbar angesehen, was durch diverse Feldstudien zumindest für technisch interessierte Nutzer bestätigt wurde (vgl. [Fra⁺01], [Hen⁺02, S. 169]).

Damit auch relativ untrainierte Endanwender das Kontextmodell in geeigneter Weise verstehen und ergänzen, aber aus Sicherheitsgründen auch inspizieren können, muß es *endnutzertauglich* gestaltet sein, worauf im folgenden eingegangen wird.

Anforderung 7 – *Endnutzertauglichkeit*

Für endnutzertaugliche, d. h. durch Endanwender ohne größeren Trainingsaufwand durchführbare Programmiermethodiken stellen ein eigenes Forschungsfeld dar, das zunehmend auf kontextbezogene Informationen angewendet wird (vgl. [New⁺02, DeSo03]). Im für diese Arbeit relevanten Bereich können die folgenden Anforderungen identifiziert werden.

Anforderung 7a – *Darstellung von Informationen auf verschiedenen Verdichtungs- bzw. Abstraktionsstufen*

Neben einer möglichst detaillierten Menge von durch Sensoren geliefertem *Basis-* oder *low-level-Kontext*, d. h. elementaren Aussagen über den Kontext als präzise Entscheidungsgrundlage, besteht seitens der Anwendungen und Endnutzer auch ein Bedarf nach *abgeleitetem* oder *high-level-Kontext*, d. h. abstrakteren, aus den einzelnen Kontextvariablen abgeleiteten Aussagen über die Verhältnisse in der aktuellen Situation (vgl. [Sch⁺99b], [GrSa01], [Lösc02, S. 54]).

Um beiden Informationsbefürfnissen gerecht zu werden, ist es letztlich notwendig, die Datenbasis sowohl in einer *low-level-* als auch einer *high-level-Perspektive* darzustellen und beide in geeigneter Weise zu verbinden. Den gleichen Gedanken äußerten HARVEL ET AL. in [Har⁺03, S. 1]⁸. Nur in einem solchen Fall können sich kontextbezogene Anwendungen ausgehend von den einfach benutzbaren, von technischen Verfahren abstrahierenden *high-level-Aussagen* und -Sichtweisen schrittweise detailliertere, feingranularere, techniknähere Darstellungen besorgen (vgl. Abb. 65 auf Seite 139). Konkret ergeben sich die folgenden Kriterien, um kontextbezogene Informationen auf verschiedenen Abstraktionsebenen darzustellen.

- Darstellung verschiedener Abstraktionsebenen einer Informationsmenge,
- Zuordnung von einzelnen Informationen zu bestimmten Abstraktionsebenen,
- Darstellung der Abhängigkeiten zwischen den Abstraktionsebenen, d. h. welche Abstraktionsebene auf welcher anderen basiert,
- Möglichkeit zur Darstellung mehrerer aufeinander basierender Abstraktionsschritte.
- Darstellung der Semantik von Abstraktion, d. h. Unterscheidung der Abstraktionsbeziehung von anderen Beziehungen zwischen Modellelementen

Anforderung 7b – *Endnutzertaugliche Programmiermethodiken*

Bei der vom *Endnutzer* durchzuführenden Überprüfung und Erweiterung von Anwendungen durch selbst erstellte Module, was als *End User Programming* bezeichnet wird, kann seitens der Anwender aus diversen Gründen weder die Bereitschaft noch die Möglichkeiten für grundlegende Lernvorgänge hinsichtlich der Benutzung eines Kontextmodells vorausgesetzt werden. Schließlich verursacht die Fortbildung trotz moderner Bildungskonzepte wie *E-Learning* durchaus signifikante monetäre, zeitliche sowie personelle Aufwendungen (vgl. Kap. 3.4.5 („E-Learning, Computer aided Instruction“, S. 68)), wobei in betrieblichen Umgebungen i. d. R. nur ein begrenztes Budget zur Fortbildung von Mitarbeitern zur Verfügung steht und Investitionen in die Ausbildung hinsichtlich neuer Detailtechnologien wie Kontextdaten deshalb wohlüberlegt werden. MERTENS erinnert in diesem Zusammenhang an die Tatsache, dass die gesamten IV-Kosten zumindest in traditionellen Wirtschaftszweigen selbst nur zwischen 1 und 4 Prozent des Umsatzes bzw. der Gesamtkosten ausmachen (vgl. [Mert95, S. 57]).

Schließlich beschränkt sich Fortbildung in diesem Zusammenhang nicht nur auf die Vermittlung von neuen Begriffen, sondern auch auf den Abgleich der durchaus unterschiedlichen *Lebenswelten*⁹ von *Modellkonstrukteuren* als i. d. R. mathematisch orientierten Informatikern und *Modellnutzern* als betriebs-

⁸„One of the challenges for ubiquitous computing is to bridge the gap between captured context data and the different levels of interpretation of captured context information needed by applications.“

⁹Der Begriff Lebenswelt wurde von HUSSERL eingeführt und bezeichnet die subjektiv von Personen erfahrene Realität als Gegenpol zur mittels wissenschaftlicher Methoden beschriebenen Welt (vgl. [Huss86]).

wirtschaftlich handelnden Managern. Beide Gruppen denken unabhängig von ihrem fehlenden Fachwissen im jeweils anderen Fachgebiet auch in unterschiedlichen *Paradigmen* (vgl. [Kuhn88b, Kuhn88a]) bzw. *Denkstilen* (vgl. [Flec80, Flec83]), haben unterschiedliche Handlungsmaximen, Interessen sowie Strategien zum Erreichen von Zielen (vgl. [Mess99]). Wenn die *Nutzer* eines Kontextmodells wie im vorliegenden Fall der Endnutzerprogrammierung auch als *Modellkonstrukteure* fungieren sollen, muß sich die Modellmethodik zumindest teilweise an die Lebenswelt der Endanwender anpassen. Zu vermeiden sind – wie bereits in Kap. 1.1 („Einordnung“, S. 1) dargestellt – allzu technische Denkstile, Paradigmen und Vokabulare, die nicht der Lebenswelt der Endanwender entsprechen (vgl. [DeSo03, S. 12]).

Letztlich hängt die Akzeptanz einer neuen Technologie in signifikanter Weise von dem Verhältnis zwischen erwartetem Nutzen und dafür aufzuwendendem Lernaufwand ab, d. h. wenn der von den Anwendern erwartete Nutzen den Aufwand für den Lernprozess nicht rechtfertigt, wird das Modell abgelehnt¹⁰. MESSER beschreibt die Akzeptanzfunktion eines Anwendungssystems aus Sicht eines Betriebswirts in diesem Zusammenhang durch die Formel $Akzeptanz = Nutzen - Lernaufwand$ (vgl. [Mess99, S. 97]).

Daher müssen sämtliche für die Benutzung des Kontextmodells notwendigen Methodiken auch für einen relativ untrainierten Endanwender einfach erlernbar und intuitiv benutzbar sein. Dazu zählen visuelle Programmiermethodiken, Programming by Demonstration und vor allem die Nutzung domänen-spezifischer Sprachen und Formalismen (vgl. [Nard93, New⁺02, Gaj⁺02, DeSo03]). Auch müssen alle technischen Elemente in geeigneter Weise eingekapselt werden, d. h. von der verwendeten Sensortechnologie und den konkreten Auswertungsverfahren sollte möglichst abstrahiert werden (vgl. [ScGe01]). Diese Anforderung steht nicht im Widerspruch, sondern orthogonal zu dem Bestreben, möglichst formalisierte, maschinenlesbare Datenformate zu erstellen, welche für den gewünschten Automatisierungsgrad innerhalb der Kontextdatenbasis benötigt werden. Es ergeben sich die folgenden Kriterien.

- Das Modell muß in einer endnutzertauglichen Weise dargestellt werden können,
- Das Funktionsprinzip des Modelles darf nicht zu kompliziert sein,
- Die Komplexität der Kontextinformationsbasis muß durch das Modell angemessen reduziert werden können.

Anforderung 7c – *Problemspezifische Sichtweisen auf das Realweltmodell*

Wie in diesem Anforderungskomplex bereits dargestellt wurde, sollten die in Computeranwendungen verwendeten Modelle durch einen Endanwender möglichst intuitiv benutzbar, leicht erlernbar und in praktischen Szenarien unkompliziert anwendbar sein. Dazu notwendig ist die Benennung einzelner Modellelemente in der Terminologie der jeweiligen Anwendungsdomänen. Im Gegensatz zu den in Anf. 6k („Synonyme Benennung von Topics“) geforderten synonymen Namen sind an dieser Stelle nicht globale, sondern anwendungs- bzw. problemspezifische Modellelemente, die nur hinsichtlich bestimmter Problemfälle relevant sind und ansonsten eher verwirrend wirken, gemeint.

Einzelne Anwendungen sind dabei i. d. R. nur an wenigen, problemspezifischen Kontextinformationen und nicht an der gesamten Kontextinformationsbasis, welche im praktischen Einsatz genau wie *Unter-*

¹⁰Als Beispiel für eine in der Praxis nicht akzeptierte Technologie gelten Petri-Netze, die zwar relativ einfach zu implementieren, aber für den Endnutzer letztlich schwer zu verstehen und zu verwenden sind (vgl. [Mess99, S. 97]).

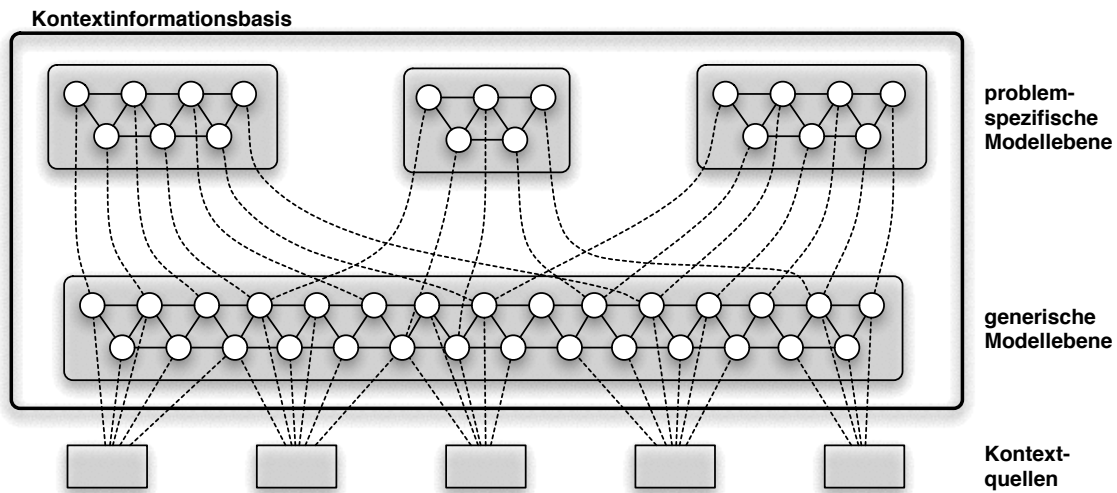


Abbildung 19: Abgegrenzte, subjektive Sichtweisen auf die Datenbasis

nehmensweite Datenmodelle schnell tausende untereinander vernetzter Objekte enthalten und damit sehr unübersichtlich werden kann, interessiert (vgl. [Hanf91, Mei⁺91, Müll89]). Abb. 19 visualisiert, wie bestimmte Informationen aus einer generischen Kontextdarstellung entnommen und zu jeweils problem-spezifischen Kontextmodellen bzw. Projektionen auf das generische Kontextmodells kombiniert werden (vgl. Anf. 6c („Explizite Ontologie“)).

Abb. 20 stellt Abb. 19 auf einer höheren Abstraktionsstufe dar. Hier wird sichtbar, dass diese problem-spezifischen Sichtweisen auf das generische Kontextmodell durch mehrere Kontextverbraucher verwendet werden können, um bestimmte Funktionalitäten gezielter als dies mit dem generischen Kontextmodell möglich ist, zu adressieren.

Die Kontextinformationsbasis sollte also problemspezifische Sichtweisen auf die Realwelt erlauben, um trotz eines umfangreichen Informationsbestandes eine einfache Implementation und Modifikation von kontextbezogener Funktionalität durch einen problemspezifischen Zugriff auf Kontextinformationen

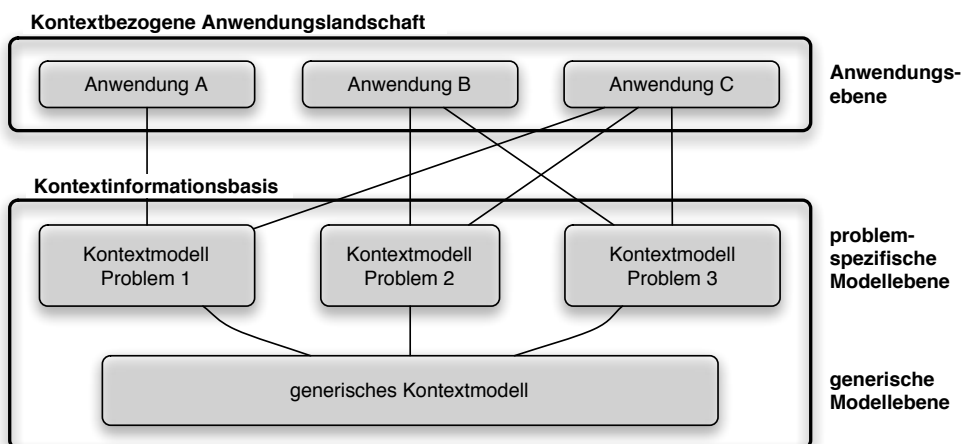


Abbildung 20: Wiederverwendbare problemspezifische Sichtweisen

zu ermöglichen. Da einzelne Informationen dabei je nach Problemstellung durchaus unterschiedlich bewertet werden können und Details in diesem Zusammenhang nicht global, sondern nur lokal sichtbar sein sollen, ist es notwendig, die problemspezifischen Sichtweisen von der restlichen Datenbasis abzugrenzen. Insgesamt ergeben sich die folgenden Kriterien.

- das Kontextmodell muß Teile des Modells in einer separaten Form darstellen können
- diese Teile müssen in einer anderen Ontologie (Begrifflichkeit, Sprache) darstellbar sein
- die Darstellung muß ggf. in einer anderen Abstraktionsstufe darstellbar sein
- Informationen in den Teilmodellen müssen mit Informationen im generischen Modell verknüpft sein
- die separate Sichtweise muß voll benutzbar, d. h. für Lese- und Schreiboperationen verfügbar sein

2.4.7 Übersicht der Anforderungen

Die gesamten Anforderungskomplexe und darin enthaltenen Anforderungen sind in Tab. 7 auf der nächsten Seite noch einmal in gesammelter Form dargestellt.

1	Allgemeine Verwendbarkeit
1a	Formalisierungsgrad
1b	Modularität
1c	Verteilte Darstellung und Speicherung
1d	Unabhängigkeit von Programmiersprachen
1e	Flexible Adressierungsmöglichkeiten
2	Multidimensionalität
2a	Separate Strukturierung von Datendimensionen
2b	Kombination separat strukturierter Datendimensionen
3	Dynamik
3a	Dynamische Elemente
3b	Generalisierende Ontologien
3c	Darstellung der Historie von Kontextvariablen
4	Datenqualität
4a	Datenqualitätsdeskriptoren
4b	Datentypspezifische Qualitätsdeskriptoren
5	Schutz der Privatsphäre
5a	Datenschutzdeskriptoren
5b	Darstellung von Akteuren
5c	Vertrauensstufen
6	Integration heterogener Kontextinformationen
6a	Realweltobjektbezogene Struktur
6b	Eineindeutige Realweltobjektmodelle
6c	Explizite Ontologie
6d	Darstellung von Kontextquellen
6e	Inhaltliche Darstellung von Kontextvariablen
6f	Verbindung vom Kontextmodell zu den Originaldaten
6g	Fusion von Kontextvariablen
6h	Integration von Methodik in das Datenmodell
6i	Darstellung des Datenpfades
6j	Einschränkung der Sichtbarkeit von Konzepten
6k	Synonyme Benennung von Topics
7	Endnutzertauglichkeit
7a	Darstellung verschiedener Abstraktionsstufen
7b	Endnutzertaugliche Programmiermethodiken
7c	Problemspezifische Sichtweisen

Tabelle 7: Anforderungen an Integrationsmodelle für kontextbezogene Informationen

3 Betriebswirtschaftliche Potenziale umfassender Kontextsichtweisen

*Our real challenge is not to build intelligent systems,
but to help corporations and governments built
intelligent organizations.*

D. Bobrow, AAAI News

In diesem Kapitel werden die wirtschaftsinformatischen und wirtschaftlichen Effekte einer umfassenden Sichtweise auf den Kontext sowie die in diesem Zusammenhang auftretende Probleme skizziert. Es wird gezeigt, dass eine umfassende Kontextinformationsbasis eine wichtige und vielseitig nutzbare Resource darstellt, die für moderne, eng in die Realweltabläufe integrierte Unternehmensanwendungen von grundlegender Bedeutung ist, denn generell sind an die aktuelle Situation angepasste Funktionalitäten ohne Kenntnis des betreffenden Kontextes seitens der maschinellen Anwendungssysteme nur sehr bedingt realisierbar. Abzugrenzen von den vergleichbaren, jedoch auf normale Kontextinformationen bezogenen Untersuchungen von FLEISCH und MATTERN ist dieses Kapitel durch den Bezug auf die in Kap. 1.2.3 („Zielstellung der Arbeit“, S. 9) beschriebenen umfassenden Kontextinformationsbestände, die erst durch die Kombination von diversen Kontextmodellen entstehen (vgl. [FIDi03, Matt04]).

Der Wert des in dieser Arbeit entwickelten anwendungsübergreifenden Darstellungs- und Integrationsmodelles für Kontextinformationen besteht in dieser Hinsicht darin, existierende Kontextinformationsbestände nicht nur durch Kombination mit weiteren Kontextinformationen qualitativ und quantitativ aufzuwerten, sondern die integrierte Kontextinformationsbasis in passender Form auch für mehrere Anwendungsfelder bereitzustellen. Dadurch können viele der nachfolgend beschriebenen Anwendungen mit im Vergleich zu herkömmlichen Verfahren deutlich verringertem Aufwand kontextsensitiv gemacht werden, indem ihnen der Zugriff auf bereits existierende, aber durch viele Anwendungen zunächst nicht verwendbare Kontextinformationen ermöglicht wird und so eine eigene Kontextbeschaffung durch die betreffenden Anwendungen unnötig wird.

Neben der Darstellung typischer Verwendungsformen von umfassenden Kontextinformationen in betrieblichen Informations- und insbesondere Anwendungssystemen werden in diesem Kapitel potentielle Anwendungsfelder für die sich daraus ergebenden kontextsensitiven Konzepte sowie die dabei auftretenden wirtschaftsinformatischen, wirtschaftlichen und technologischen Effekte umrissen.

3.1 Modellseitige Effekte durch die Kombination von Kontextinformationen

Durch die Kombination von Kontextinformationen aus verschiedenen Kontextquellen und -dimensionen verbessert sich die Qualität der Kontextdatenbasis hinsichtlich der folgenden Aspekte.

Genauigkeit von Kontextvariablen. Die Genauigkeit einzelner Sensoren ist begrenzt. Zur Erreichung darüber hinausgehender Genauigkeiten müssen die Ergebnisse verschiedener Sensoren oder Meßverfahren kombiniert werden. Zum Beispiel können die Ungenauigkeiten bei der Lokalisierung von Realweltobjekten drastisch reduziert werden, wenn die Messwerte verschieden positionierter Sensoren mitein-

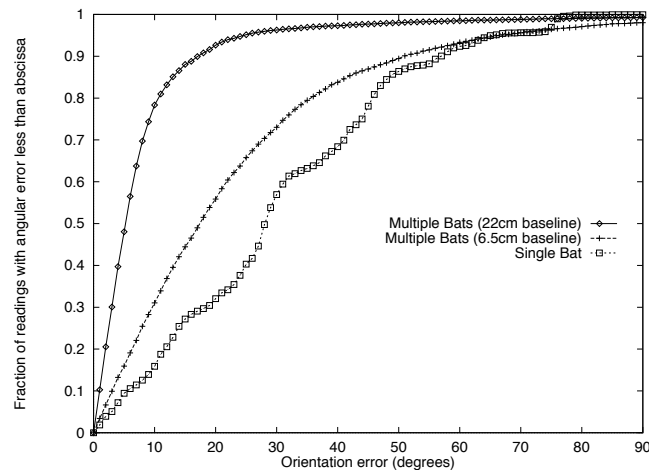


Abbildung 21: Genauigkeit der Bestimmung der räumlichen Orientierung von Endgeräten i. A. von Anzahl und Abstand der dazu verwendeten Sensoren ([Har⁺01, S. 3])

ander kombiniert werden, wie dies von HARTER in [Har⁺01, S. 3] demonstriert wurde (vgl. Abb. 21). Der Grad der Verbesserung der Meßgenauigkeit ist dabei nicht nur von der Anzahl der verwendeten Sensoren, sondern auch von deren Anordnung und Entfernung abhängig.

Ebenso können verschiedene Verfahren miteinander kombiniert werden, etwa könnte das mittels eines globalen Lokalisierungssystems identifizierte Fahrzeug, in welchem sich der Nutzer befindet, gezielt nach dessen Position innerhalb des Fahrzeugs befragt und auf diese Weise globale und lokale Ortskoordinaten miteinander kombiniert werden, was für bestimmte Anwendungen wie für ortsbezogene Assistenzsysteme, die für Fahrer und Beifahrer verschiedene Interaktionsmodalitäten verwenden, durchaus relevant ist (vgl. Kap. 6.2.2 („Messenger-VUI“, S. 179)).

Umfang des Kenntnisstandes über Realweltobjekte. Da einzelne Kontexterfassungssysteme auf die Erfassung einzelner Kontextdimensionen spezialisiert sind und daher auch nur Informationen hinsichtlich dieser Kontextdimension bereitstellen können, eröffnet erst die Kombination der Messwerte verschiedener Kontextquellen eine umfassende, d. h. verschiedene Aspekte der Realität beinhaltende Kenntnis über Realweltobjekte (vgl. Kap. 2.2 („Arbeitsthese“, S. 22)).

Lücken im Kontextmodell. Kontextdaten sind fast immer unvollständig. Dies liegt nicht nur an technischen Grenzen real existierender Kontexterfassungssysteme, die nicht immer und überall gleichermaßen zuverlässig funktionieren. Eine Lokalisierung per GPS ist etwa nur außerhalb von Gebäuden möglich, während indoor-Lokalisierungsverfahren, die etwa Infrarotbaken in dazu präparierten Zimmern zur Erkennung ihres Ortes nutzen, nur innerhalb von Gebäuden funktionieren. Daher müssen in der Praxis funktional redundante Verfahren kombiniert werden, um die Zuverlässigkeit der Kontextinformationsversorgung, welche für robuste kontextsensitive Anwendungen notwendig ist, in allen denkbaren Situationen, in diesem Fall sowohl innerhalb als auch außerhalb von Gebäuden, sicherzustellen (vgl. [Hong01, S. 1]).

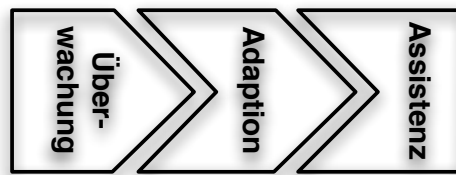


Abbildung 22: Aufgabenkontinuum kontextbezogener Anwendungssysteme

3.2 Nutzungsformen von Kontext

Kontextinformationen werden von kontextsensitiven Werkzeugen verwendet, um ihr eigenes Erscheinungsbild, ihr Verhalten bzw. ihre Funktionalität sowie ihre Daten und Dokumente an die aktuelle Situation anzupassen. Dadurch soll das Leben und Handeln der menschlichen Benutzer sicherer, komfortabler und effizienter gestaltet werden (vgl. [Hong01, S. 2]). Diese Funktionen können in einem in Abb. 22 dargestellt Kontinuum angesiedelt werden, welches aus drei unscharf voneinander getrennten Segmenten besteht. Kontextsensitive Funktionalität erstreckt sich dabei ausgehend von einer *Überwachung* der Realwelt über Funktionen zur *Adaption* von Objekten bis hin zur *Assistenz* menschlicher Nutzer. Diese Aspekte werden im folgenden detaillierter vorgestellt.

3.2.1 Überwachung

Im Gegensatz zur *Betriebsdatenerfassung* (BDE), die vor allem Informationen aus dem internen Produktionsplanungs- und Steuerungssystem (PPS) zur Steuerung der Produktionsprozesse verwendet, ermöglicht die Berücksichtigung von Kontextinformationen die *realweltbezogene* Überwachung von Bereichen, d. h. eine Überwachung von Prozessen anhand von deren realweltbezogenen Auswirkungen. Diese automatisierte Kontrolle kann etwa in sicherheitskritischen oder hochautomatisierten Produktionsbereichen gewünscht sein. Der Einsatz von Technik hat gegenüber einer manuellen Überwachung durch menschliche Aufgabenträger den Vorteil, die betreffenden Informationen in einer für Menschen nicht möglichen Aktualität, Qualität und Quantität, d. h. in einer größeren Schnelligkeit, Genauigkeit und Menge erfassen zu können bzw. für Menschen überhaupt nicht sichtbare Realweltparameter, etwa den Grad an radioaktiver Strahlung, zu erfassen. Kontextsensitive Technologie erweitert dabei das Spektrum der maschinell erfassbaren Realweltparameter im Vergleich zu herkömmlichen Verfahren.

3.2.2 Adaption

Unter *Adaption* wird die Spezialisierung von Dokumenten und Diensten an eine bestimmte Umgebung verstanden (vgl. [Spri04, S. 9]). Während *adaptierbare* Systeme lediglich eine manuelle Konfiguration durch den Nutzer erlauben, können sich *adaptive* Systeme selbsttätig an die jeweilige Umgebung anpassen (vgl. [Manb00]). Im Folgenden werden die wichtigsten *Adaptionsobjekte* genannt.

Adaption von Anwendungen. Kontextbezogen adaptiert werden können verschiedenste Aspekte und Teile von Anwendungssystemen. Die betrifft *Anwendungsdaten*, etwa deren kontextbezogene Vertei-

lung im Netzwerk mittels kontextgesteuerter Replikation, Zwischenspeicherung oder ein Vorabladen auf mobile Endgeräte bei Verfügbarkeit günstiger Verbindungen, die *Nutzerschnittstelle* sowie die *Struktur* von Anwendungen selbst, etwa durch Verlagerung ressourcenintensiver Anwendungskomponenten auf verschiedene Endgeräte dank mobilem Code (vgl. [Sch⁺93], für eine Übersicht über mobile Codesysteme vgl. [Spri04, S. 52 f]). Die Adaption von Anwendungen wird i. d. R. durch adaptive Anwendungskomponenten realisiert, welche eine Adaptionsschnittstelle zur Steuerung der Adaption der Komponente mittels Kontextinformationen aufweisen. Solche adaptiven Anwendungskomponenten wurden u. a. von SPRINGER entwickelt und beschrieben (vgl. [Spri04]).

Adaption von Dokumenten. Neben Anwendungsdaten können auch *Dokumente* adaptiert werden. Aufgrund ihrer flexiblen Struktur werden dazu oft *adaptive Hypertexte* verwendet, wobei deren Inhalt, Navigationsstruktur oder Präsentation hinsichtlich der gegebenen Situation angepasst werden, d. h. irrelevanter Inhalt wird ausgeblendet, bestimmte Links bzw. Linktypen (etwa Erläuterungs- bzw. Navigationslinks) können i. A. vom aktuellen Nutzer und dessen Kenntnisstand angepasst, verborgen oder mit verschiedenen Kommentaren versehen werden und das Dokument wird in verschiedenen Schriftgrößen, Farbschemata oder Modalitäten gemäß den Präferenzen und Fähigkeiten des Benutzers dargestellt (vgl. [BrMa02, Kapl93, Brus98, Boyl94]). Auch multimediale Inhalte können i. A. verschiedener technischer und personeller Umstände adaptierbar gestaltet werden (vgl. [Fia⁺03]).

Adaption von Daten. Die vom menschlichen Benutzer zur Abwicklung der Geschäftsprozesse abgefragte Informationsmenge kann durch eine selbstständige Beschaffung der benötigten Informationen mittels zusätzlicher kontextbezogener Eingabekanäle in Anwendungen verringert werden. Dabei werden nicht nur Daten aus der Realwelt, sondern auch aus anderen Anwendungen beschafft (vgl. Kap. 2.4.5 („Integration heterogener Kontextinformationen“, S. 35)). Darüber hinaus können Anwendungsdaten vor ihrer Herausgabe an einen bestimmten, kontextsensitiv identifizierten Benutzer anonymisiert, verrauscht oder abstrahiert werden, um geltende Datenschutzrichtlinien einzuhalten.

Adaption von Umgebungen. Auch die physikalische Umgebung des Nutzers kann kontextsensitiv adaptiert werden. Dies betrifft nicht nur kontextsensitive Gebäudesteuerungen (vgl. Kap. 3.4.9 („Gebäudemanagement“, S. 73)), sondern auch weitergehende Konzepte aus dem *Ubiquitous Computing*, wobei eine durch eingebaute Elektronik angereicherte Umgebung erschaffen werden soll, die sich auf den aktuellen Nutzer und seine Aufgaben einstellt (vgl. Kap. 3.3.3 („Ubiquitous/Pervasive Computing, Ambient Intelligence“, S. 56), [VaBo93, Hong01]).

In jedem dieser Anwendungsfelder von Adaption können die im folgenden dargestellten *Adaptionsoperationen* durchgeführt werden.

Durch das **Weglassen irrelevanter Komponenten** werden nur die in der aktuellen Situation wirklich benötigten Komponenten verwendet bzw. dargestellt.

Ein **Austausch von Komponenten** wird möglich, wenn für jede Komponente eine Reihe von verschiedenen Alternativen existiert, wobei immer die für die augenblickliche Situation geeignetste Variante verwendet wird (vgl. [Fia⁺03]). Ebenso können Komponenten durch Adaption der Komponenten selbst an die aktuelle Situation angepasst werden.

Erweiterung von Komponenten. Die Funktionalität von Komponenten wird mittels zusätzlich verfügbarer Realweltinformationen erweitert, d. h. es werden Funktionen dieser Komponenten möglich, die ohne Realweltinformationen nicht möglich sind.

SPRINGER begründete die Notwendigkeit einer systemweit gültigen Kontextinformationsbasis mit ihrer Funktion als Grundlage für eine alle Anwendungen umfassende *kooperative Adaption*, welche die zur Verfügung stehenden Ressourcen mittels eines gesamtheitlichen, d. h. sämtliche adaptiven Komponenten umfassenden Adaptionsverhaltens bestmöglich ausnutzen soll. Dadurch sollen gegensätzliche oder ineffektive Adaptionen einzelner Teilsysteme vermieden werden, die etwa darin bestehen können, dass bei einem auftretenden Ressourcenengpaß eine kontextbewusste Komponente diese Ressource durch Adaption von sich selbst freigibt, worauf diese Ressourcen sofort durch eine nicht kontextsensitive Komponente alloziiert und damit der Effekt der Adaption zunichte gemacht wird (vgl. [Spri04, S. 63 f]).

3.2.3 Assistenz

Während *Informationssysteme* den Nutzer mit kontextbezogenen Informationen versorgen, um dessen eigene Aktionen zu unterstützen, gehen *Assistenzsysteme* einen Schritt weiter und erledigen bestimmte Aktionen, die im Kontext des Nutzers erforderlich werden, selbsttätig (vgl. [FeSi98, S. 104 f]). Dabei können das sogenannte *proaktive* Verhalten, welches die zukünftigen Tätigkeiten des Nutzers antizipiert und bestmöglich vorbereitet, die Unterstützung des Nutzers bei der tatsächlichen Durchführung der Tätigkeiten sowie die *reaktive* Unterstützung des Nutzers durch Ausführung von seinen Tätigkeiten nachgelagerten Assistenzfunktionen unterschieden werden.

Dabei kann wiederum zwischen der Unterstützung und der kompletten Übernahme der Tätigkeiten des Nutzers unterschieden werden. Eine *Unterstützung* kann etwa darin bestehen, die in der aktuellen Situation notwendigen Informationen automatisch an den Nutzer auszugeben, wodurch dieser in die Lage versetzt werden soll, sich voll auf die *Nutzung* der Informationen bei seinen eigentlichen Geschäftstätigkeiten und nicht auf die eigentlich nicht zielführende *Recherche* dieser Informationen zu fokussieren. Bei der *Übernahme* von Tätigkeiten des Nutzers kann zwischen deren vollkommen selbsttätiger Ausgeführtung und deren Vorbereitung unterschieden werden wobei vorbereitete Aktionen vom Nutzer bei Bedarf nur noch ausgelöst werden müssen (vgl. [Lösc02, S. 10]).

3.3 Kontextbezogene betriebliche Anwendungssysteme

Moderne betriebliche Anwendungssysteme als automatisierter Teil der betrieblichen Informationssysteme stellen nicht länger nur die Systeme dar, mit denen die Angestellten in Unternehmen arbeiten. Moderne Backendsysteme werden mittels einer Reihe von Frontends sowohl durch die Angestellten als auch direkt durch Endkunden der Unternehmen zugegriffen, was unter dem Schlagwort *Business-to-Consumer* zusammengefasst wird. Moderne Geräte- und Kommunikationstechnologien ermöglichen dabei die zunehmende Integration der Frontends betrieblicher Anwendungssysteme in die unmittelbare Umgebung der Mitarbeiter und Kunden und erreichen dadurch eine enge Integration von betrieblichen Prozessen mit den in der Realwelt ablaufenden und durch die Unternehmensanwendungen zu unterstützenden Prozessen.



Abbildung 23: Wearable Computing im betrieblichen Außendienst

Daher spielen mobile Unternehmensanwendungen, also zunehmend in die Umwelt der sie benutzenden Personen integrierte Systeme, eine zunehmend bedeutende Rolle zumindest in gewissen Unternehmensbereichen (vgl. u. a. [Schw03]). Obwohl Kontextinformationen von sämtlichen Arten betrieblicher Anwendungssysteme verwendet werden können, sind in die Realwelt integrierte Anwendungssysteme in besonderem Maße auf Kontextinformationen angewiesen, um ihre Funktionalität an der aktuellen Situation der Benutzer zu orientieren. Dabei kommen zunehmend auch die im folgenden etwas detaillierter beschriebenen modernen kontextbezogenen Technologien wie Wearable oder Pervasive Computing zum Einsatz, wobei Konzepte zu deren Integration mit wirtschaftlichen Prozessen bereits seit einigen Jahren im Brennpunkt wissenschaftlicher Untersuchungen stehen (vgl. [FIDi03, Matt04]).

3.3.1 Wearable Computing

Unter *Wearable Computing* wird die Integration leichtgewichtiger, tragbarer Computersysteme oder zumindest von deren Schnittstellen (d. h. Kameras, Sensoren, Aktoren, head-mounted displays und Miniaturtastaturen) in die Kleidung oder kleidungsähnliche Gebrauchsgegenstände wie Brillen (für Displays) oder Ohrhörer (für Lautsprecher) verstanden (vgl. Abb. 23, [Pasc98, Star99b, BiSt99]). Durch ihre mobile Konzeption sind Wearable Computer in einem wesentlich stärkeren Maß geländetauglich, d. h. schmutz- und wasserresistent sowie auch bei starkem Sonnenlicht ablesbar. Im Gegensatz zu PDAs, die explizit in der Hand gehalten werden, sind Wearable Computer dafür entworfen, direkt am Körper getragen genutzt und durch sprachliche Kommandos oder zumindest nur mit einer Hand gesteuert zu werden (vgl. [GeTe01, Lyo⁺03]). Im Gegensatz zum Desktop-Paradigma, bei dem der Computer die klassischen Arbeitsabläufe (z. B. Papierdokumente, Notizblock, Briefpost, Telefon) schrittweise ersetzt und die Arbeitsleistung daher im Extremfall nur noch durch Interaktion mit einem Computer erbracht wird, *ersetzen* Wearable Computers die nach wie vor in der Realwelt stattfindenden Arbeitsabläufe nicht, sondern *unterstützen* lediglich die Erfüllung dieser primären Aufgabenstellungen, was auch als *Arbeiten mit* anstatt *Arbeiten am* Computer bezeichnet wird (vgl. [Boro02]).

Wearable Computing ermöglicht ein sehr fortschrittliches Benutzungskonzept für stark in die Realwelt integrierte Informations- und Kommunikationssysteme, welches in mehrfacher Hinsicht vorteilhaft wirkt. Zum einen können die direkt in das Geschehen involvierten tragbaren Computersysteme ein breites Spektrum von im Alltag des Nutzers auftretenden (Kontext-)Information direkt erfassen und für die im weiteren Verlauf dieser Arbeit vorgestellten kontextbezogenen Assistenzdienste verfügbar machen. Beispiele stellen die Erkennung von Textstücken durch Abfotografieren und Schrifterkennung

(vgl. u. a. [Sta⁺98]), unterwegs getroffene Personen per mobiler Gesichtserkennung (vgl. [Wre⁺95, Sti⁺98, SiSt01b]) und gesprochene Informationen per Spracherkennung dar (vgl. [HiSc92, Star02b]). Zum anderen können auf diesen Informationen basierende Assistenzdienste durch Wearable Computing auch gleich in der betreffenden Situation zugänglich gemacht werden, denn es erlaubt die jederzeitige, alltägliche, pervasive Interaktion von Nutzern mit IuK-Systemen, ohne dabei auf den Bedienungsoverhead explizit zu benutzender – d. h. in einer möglicherweise nicht dafür verfügbaren Hand zu haltender – Geräte und die damit einhergehende Unterbrechung der Arbeitsabläufe sowie den zur Bedienung eines Computers notwendigen kognitiven Overhead angewiesen zu sein. Dies wirkt sich besonders in unübersichtlichen Anwendungsszenarien wie dem Warten komplexer technischer Geräte aus, die aufgrund der hohen Ausfallkosten oft zeitkritisch sind. Dabei sind die Augen und Hände des Arbeiters auf das zu bearbeitende Objekt fokussiert und stehen daher nicht für eine Interaktion mit einem PDA zur Verfügung. In solchen Situationen ist meist auch gar kein vollständiger Zugriff auf das betriebliche Informationssystem gewünscht, sondern nur eine Interaktion mit anderen Teammitgliedern oder der Zentrale sowie die Kenntnis der für den aktuellen Arbeitsschritt benötigten Informationen, etwa eine Seite aus dem Wartungshandbuch, die dem Nutzer zur richtigen Zeit akustisch oder optisch mitgeteilt werden kann oder sogar nur die Stärke in Newtonmetern, mit der eine bestimmte Schraube in der Turbine festgezogen werden soll (vgl. [O.V.04g]).

Wearable Computing ermöglicht also nicht nur einen permanenten Zugriff auf die Funktionen von Unternehmensanwendungen sowie eine vereinfachte Kommunikation zwischen menschlichen Aufgabenträgern in mobilen Einsatzszenarien, sondern macht es darüber hinaus möglich, Informationen aus den betrieblichen Anwendungssystemen direkt in den realweltbezogenen Arbeitsprozess zu integrieren, ohne explizit einen Computer benutzen zu müssen (vgl. [Kor⁺98, Star99a, Boro02]). Auf diesen Aspekt wird im nächsten Abschnitt konkreter eingegangen.

Problematisch an Wearable Computing ist, dass die Aufmerksamkeit des Benutzers durch visuelle oder akustische Signale von der Realität abgelenkt wird, wodurch diese tragbaren Computersysteme nicht in jeder Situation eingesetzt werden können. Beim Autofahren sollten z. B. keine visuellen Interaktionen mit dem Nutzer durchgeführt werden, da diese einen Teil seines Gesichtsfeldes verdecken und so seine Fahrtauglichkeit beeinträchtigen können (vgl. [Star02a]).

3.3.2 Augmented Reality

Sämtliche Arten von Computerbildschirmen stellen im Prinzip immer noch elektronifiziertes Papier, d. h. ein zweidimensionales Medium auf dem etwas dargestellt wird, dar. Die Wahrnehmung und damit die Erlebniswelt des Menschen spielt sich jedoch in drei Dimensionen ab. Den Idealfall der Informationsbereitstellung stellt also nicht die Darstellung von Informationen auf einem (ggf. tragbaren) Computerbildschirm dar, sondern die räumlich korrekte Integration dieser Informationen in die dreidimensionale Wahrnehmung des Nutzers. Diese Integration von virtuellen Informationen und realen Objekten wird als *Augmented Reality* (elektronisch angereicherte Realität) bezeichnet und durch halbdurchlässige, in Brillen integrierte Computerdisplays realisiert, die zu den Objekten, die der Nutzer ansieht, stereoskopische, räumlich korrekt in die Weltsicht des Nutzers integrierte Informationen anzeigen und so die

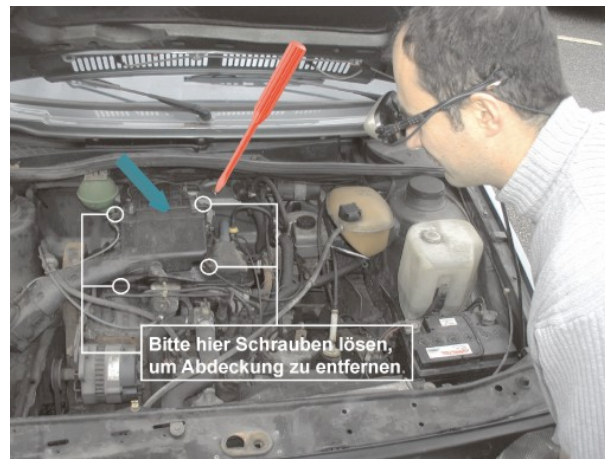


Abbildung 24: Augmented Reality bei der Fahrzeugwartung

Wahrnehmung der realen Welt durch den Nutzer mit Informationen aus der virtuellen Welt anreichern (vgl. Abb. 24, [Sta⁺97, Azum01, Sta⁺03]).

Durch Augmented Reality können Informationen aus dem betrieblichen Informationssystem bei Bedarf direkt in das Sichtfeld des Nutzers eingeblendet werden. Der Nutzer muß dazu seine aktuelle Tätigkeit nicht unterbrechen, um auf ein explizites Display oder ein Handbuch zu blicken und kann sich die ganze Zeit voll auf die zu bearbeitende Aufgabe konzentrieren. Ferner wird dadurch die Informationsflut dahingehend verhindert, dass die Informationen nur an den Stellen eingeblendet werden, wo sie auch benötigt werden. Damit wird die Realität als Zugriffsindex zur Adressierung bzw. zum Filtern der benötigten Informationen verwendet. Dies wird in Konzepten zur realweltbasierten Informationsadressierung verwendet (vgl. Kap. 4.1.1 („Stellvertreteransatz“, S. 97)).

3.3.3 Ubiquitous/Pervasive Computing, Ambient Intelligence

In Zukunft werden wir verschiedene Computer, die zusammen eine intelligente Umgebung bilden, genauso verwenden wie wir heute verschiedene kooperierende Anwendungen in expliziten Computersystemen verwenden. Diese von WEISER Anfang der 1990er Jahre entwickelte die Vision des *Ubiquitous Computing* als der dritten fundamentalen Generation von Computersystemen basiert auf der Extrapolation des in Abb. 25 auf der gegenüberliegenden Seite dargestellten Trends der Nutzerzahlen pro Computersystem. Ausgehend von sehr kostspieligen *Großrechnern*, die von ganzen Expertengruppen betrieben sowie von vielen Nutzern gleichzeitig mittels Terminals benutzt wurden über *Personal Computer*, die zu je einem Nutzer gehören, wird die Entwicklung basierend auf der Annahme einer ständig steigenden Leistungsfähigkeit bei stark fallenden Preisen hin zu einem Szenario extrapoliert, in dem jeder einzelne Nutzer viele computergestützte Geräte verwendet. Diese vielen Geräte werden nicht in Form expliziter PC-Systeme verwendet, sondern in die physikalische Umgebung, d. h. die einzelnen Realweltobjekte integriert und miteinander vernetzt, wodurch sie eine *intelligente Umgebung* bilden, mit der intuitiv und in realweltbezogenen Paradigmen interagiert wird (vgl. [Str⁺98]). Dies bedeutet, dass der Zugang zum elektronischen Universum im ubiquitären Szenario nicht mehr nur über einen expliziten Computer, sondern über eine Vielzahl von elektronisch aufgewerteten, miteinander vernetzten Gebrauchsgegenständen

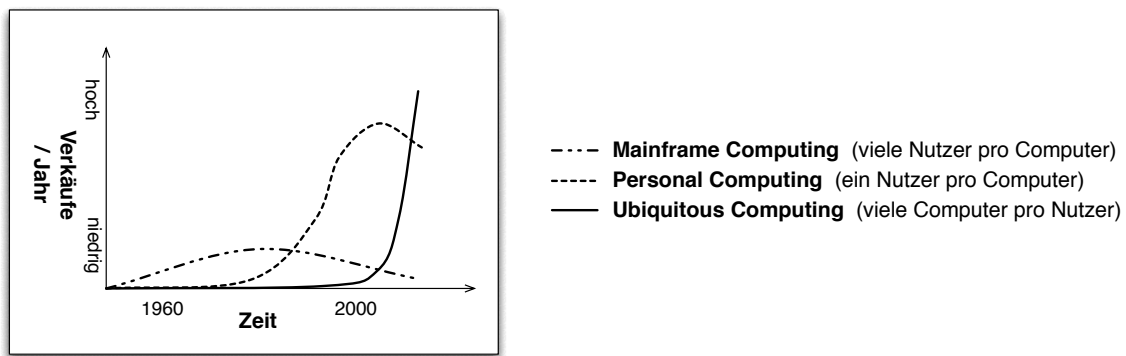


Abbildung 25: Entwicklung der kumulierten Verkaufszahlen von Rechnersystemen der drei Computing-Paradigmen (vgl. [Weis91])

realisiert wird und dadurch Dienste ermöglicht werden, die mit heutigen PCs nicht möglich bzw. sinnvoll sind (vgl. [Weis91, Norm98]). Einen weiterführenden Überblick über das Thema gibt GELLERSEN in [Gell99].

Während beim *Ubiquitous Computing* vor allem das „unsichtbar werden“ der Computertechnik, also die Integration der Computertechnik in Realweltobjekte im Vordergrund steht (vgl. [Weis91], auch als *Embedded Computing* bezeichnet), beschäftigt sich *Pervasive Computing* mit kleinen mobilen Endgeräten, die immer noch wie technische Geräte aussehen und direkt durch Endanwender für den permanenten mobilen Informationszugriff in allen Lebenslagen verwendet werden (vgl. [SaMu03, McCr00]).

Die vorwiegend in Europa entwickelte und als eine der zentralen Zukunftstechnologien für Europas Industrie angesehene Vision von *Ambient Intelligence* ermöglicht ebenfalls die einfache und natürliche Interaktion mit netzwerkbasierenden Diensten durch Integration dieser in alltägliche Umgebungen und erweitert damit ebenfalls die Lebens- und Arbeitsumgebung von Menschen durch intelligente Funktionen (vgl. [Wis⁺98, RoCl02, Schu04]).

3.4 Anwendungsfelder von Kontextinformationen in betrieblichen IuK-Systemen

Im folgenden werden betriebliche Anwendungsbereiche vorgestellt, welche durch die Versorgung von maschinellen und personellen Aufgabenträgern mit umfassenden, detaillierten und strukturierten Kontextinformationen in funktioneller Hinsicht erweitert werden können. Diese Anwendungsfelder beziehen sich nicht nur auf die eben beschriebenen mobilen kontextbezogenen Anwendungssysteme, sondern können aufgrund des Querschnittscharakters des Kontextinformationsmanagements in prinzipiell allen betrieblichen Anwendungsfeldern angesiedelt werden, wobei sich sinnvolle Anwendungen für Kontextinformationen dabei sowohl auf der Ebene der *operativen Systeme* als auch in administrativen *Planungs- und Kontrollsystemen* finden (vgl. [Mert03, S. 50]).

3.4.1 Informationsmanagement, Information Support

Unter *Information Support* als Teilbereich des Informationsmanagements wird die Befriedigung des objektiven und subjektiven Informationsbedarfes von betrieblichen Aufgabenträgern, d. h. die Versorgung betrieblicher Aufgabenträger mit den zur Aufgabenerfüllung notwendigen Informationen verstanden. Information Support basiert auf der Kenntnis der aktuellen Aufgabenstellungen des Nutzers und der dafür benötigten Informationen (vgl. [BuHa00, Maus01]). Dazu werden neben den im Folgenden genauer vorgestellten *Informationsleitständen* auch auf *Taskmodellen* basierende proaktive Systeme verwendet, welche den vom Nutzer benötigten Informationsbedarf basierend auf einer Prognose seiner aktuellen Tätigkeit bestimmen und selbsttätig *just-in-time* bereitstellen.

3.4.1.1 Informationsleitstand

Primär sollen Kontextinformationen durch maschinelle Aufgabenträger, also Anwendungssysteme, verwendet werden. Diese können die betreffenden Informationen dann in einer aufbereiteten und ausgewerteten Form auch für menschliche Benutzer darstellen, z. B. in Form eines *Informationsleitstandes* (vgl. [Mert03, S. 64 ff]).

Ein solcher Leitstand hat die Aufgabe, Entscheidungsträger im betrieblichen Umfeld mit den zur Entscheidungsfindung benötigten Informationen zu versorgen. Dazu zählen neben innerbetrieblichen Kennzahlen auch außerbetriebliche Informationen und sonstige Umgebungsdaten, die relevant für die betrieblichen Entscheidungsprozesse sind und daher den Entscheidungsträgern mittels des Informationsleitstandes bekannt gemacht werden müssen (vgl. [Mert03, S. 51]). Da die von einem Informationsleitstand darzustellenden Informationen auf den Menschen und nicht auf Maschinen als Verbraucher abzielen, muß ihre Darstellung zwar gewisse Mindestanforderungen hinsichtlich Formalisierung und Strukturierung erfüllen, sollte die Kreativität der Entscheidungsträger durch eine zu trockene Darstellung jedoch nicht zu sehr einengen (vgl. [Fuc⁺99, Fuch97]).

Ein Informationsleitstand, der heute üblicherweise als Softwareanwendung realisiert wird, stellt Informationen mittels diverser Informationselemente dar und ermöglicht eine Interaktion mit dem Nutzer mittels Kontrollelementen.

Ein kontextbezogener Informationsleitstand hat im betrieblichen Alltag eine Reihe verschiedener Anwendungsmöglichkeiten. Neben der Analyse des tatsächlichen Ablaufes der theoretisch geplanten Prozesse in der Realwelt werden dadurch praxisnähere und detailliertere *Soll-Ist-Vergleiche* möglich, deren deskriptive Komponente (Ist-Zustand) sich nicht mehr auf theoretisierte, d. h. immer noch am theoretisch geplanten Ablauf orientierte Prozessdaten aus den PPS-Systemen bezieht, sondern den in der Realwelt tatsächlich stattfindenden Prozessverlauf durch Darstellung der in der Realwelt eintretenden Ausprägungen der betrieblichen Aktionen widerspiegelt und damit substanziell praxisorientierter als herkömmliche Informationsleitstände ist. Dadurch können insbesondere Fehler und Unregelmäßigkeiten bei der Ausführung von Prozessen in der Realwelt geeigneter eingegrenzt und deren Ursachen gezielter identifiziert werden.

Während *konkrete* Prozesdaten vor allem im mittleren und unteren Management eine Rolle spielen, sind die Aufgabenbereiche des betrieblichen Top-Managements auf der strategischen Ebene anzusiedeln und benötigen daher eher *abstrakte*, strategische Informationen ohne unmittelbaren Bezug zu einzelnen,

aktuell ablaufenden Realweltprozessen. Kontextinformationen bringen für das Top-Management folglich zumindest in direkter Hinsicht keinen nennenswerten Nutzen, sondern stellen eher Eingabeparameter im Rahmen weiterführender Analysefunktionen dar.

Beispiel 7 Bei der von MERTENS beschriebenen, mittels IT-Systemen erweiterten Risikoanalyse nach HERTZ wird zur kompetenten Entscheidungsfindung bei schwierigen Problemstellungen die Meinung von innerbetrieblichen Kompetenzträgern konsultiert.

Während MERTENS in diesem Zusammenhang von existierenden *Kompetenzdatenbanken* im Unternehmen ausgeht, bleibt offen, wie solche Datenbanken realistischerweise zu erstellen und vor allem aktuell zu halten sind. Die Analyse von umfassenden Kontextinformationen über die Angestellten, etwa deren Kommunikationsinhalte, Projektarbeit und Auslandsaufenthalte, ermöglicht an dieser Stelle ein automatisiertes Profiling der individuellen Kompetenzen und Erfahrungen der Mitarbeiter und kann daher die gewünschten Ansprechpartner für bestimmte Problemstellungen benennen. FENN nennt dies *Skill Mining* (vgl. [Fenn99]). Teilweise wird dieser Ansatz bereits realisiert. So extrahiert HEEREN in seinem XPERTFINDER genannten System die Gesprächsinhalte aus der EMail-Korrespondenz von Anwendern, um daraus deren jeweilige Kompetenzen und Zuständigkeiten zu prognostizieren und diese Informationen in einer zunächst anonymisierten Form zur Findung von Kompetenzträgern anzubieten (vgl. [HeSi01]). Einen ähnlichen Ansatz verwendet MATTOX in [Mat⁺99]. Weitergehende Ansätze könnten wie bereits erwähnt Informationen aus weiteren Kontextdimensionen, etwa über bisherige Mitarbeiter (d. h. Personenwissen), betriebliche Projekte (Projekt- und Fachwissen) und Einsatzorte (Ortskenntnis) auswerten. Die in dieser Arbeit entwickelten *Context Maps* würden an dieser Stelle eine Integration dieser heterogenen Nutzerinformationen und damit eine umfassende Analyse der Nutzerkompetenzen erlauben, was ebenfalls im Rahmen des betrieblichen Wissensmanagements von Bedeutung ist, worauf in Kap. 3.4.3 („Wissensmanagement“, S. 64) genauer eingegangen wird.

3.4.1.2 Kontextbezogenes Informationsmanagement

Kontextinformationen müssen durch ein Kontextinformationsmanagement nicht nur inhaltlich verarbeitet werden, sondern durch eine kontextgesteuerte *Informationslogistik* auch zur richtigen Zeit am richtigen Ort, also kontextabhängig, zur Verfügung gestellt werden (vgl. [Sonn99, S. 43], [Krcm92]). Auch für diese Entscheidungen werden wieder entsprechende Kontextinformationen etwa über die aktuelle Situation und insbesondere die dabei relevanten Themen benötigt. Ein Beispiel hierfür stellt der *Information Logistics*-Ansatz von MEISSEN dar, der Informationen basierend auf einer Analyse der Problemstellungen in der gegebenen Situation an den Nutzer zustellt (vgl. [Mei⁺04]). Auf diesen wird in Kap. 3.4.12.1 („Kontextsensitive Telekommunikation“, S. 76) genauer eingegangen.

3.4.1.3 Information Retrieval

Unter *Information Retrieval* (IR) wird das Abrufen gespeicherter Informationen oder Dokumente verstanden (vgl. [BaRi01]). Es kann manuell durch den Nutzer veranlasst oder proaktiv durch Assistenzsysteme ausgeführt werden, um zum Zeitpunkt des Informationsbedarfes bereits mit den vorab recherchierten Informationen aufwarten zu können. Dabei tauchen Kontextinformationen an zwei Stellen im

Retrievalprozess auf, zum einen als zu suchende Inhalte, zum anderen als Mittel zur Konfiguration beliebiger anderer Suchvorgänge. Da der erste Bereich bereits hinreichend durch existierende IR-Verfahren abgedeckt wird, ist an dieser Stelle vor allem der zweite Bereich von Interesse.

Information Retrieval wird i. d. R. durch Suchmaschinen realisiert, welche jedoch aufgrund ihres begrenzten Domänenwissens nicht immer das gewünschte Ergebnis finden können. SMYTH untersuchte die Erweiterung von Suchvorgängen in Suchmaschinen um Kontextinformationen, etwa hinsichtlich der Präferenzen des aktuellen Benutzers sowie der Auswertung von dessen Reaktion auf früher dargestellte Suchergebnisse. Allein durch eine Personalisierung der Suche konnte eine Steigerung der Trefferrelevanz von 45% auf 70% gegenüber der unpersonalisierten Suche erreicht werden (vgl. [Smy⁺02]). Dies wird u. a. dadurch erreicht, dass für *Einsteiger* in ein Themengebiet eher einführende Informationen in ein Themengebiet dargestellt werden, da diese Personen technische Details sowieso nicht verstehen bzw. diese Informationen nur zusammen mit angemessenen Erklärungen wünschen, während fortgeschrittene Anwender genau diese spezifischen Detailinformationen benötigen und einführende Informationen eher als lästig empfinden (vgl. u. a. [Geri00]). Ebenfalls können bereits abgerufene Informationen ausgefiltert werden, um die Treffermenge auf die Informationen zu beschränken, die der Nutzer noch nicht kennt (vgl. [Dey98, Maus01]).

Bedingt durch den assoziativen Charakter des menschlichen Erinnerungsvermögens, welches von TULVING als *episodic memory* bezeichnet wird, spielen die *Umstände*, unter denen Ereignisse auftreten, eine erhebliche Rolle beim menschlichen Information Retrieval, denn Kontext spielt bei der Strukturierung des menschlichen Gedächtnisses eine entscheidende Rolle (vgl. [Tulv83, Lam⁺94, Srin97]). Besonders wenn nur vage Erinnerungen bestehen, die nicht mehr direkt benannt werden können, adressiert der Mensch Gedächtnisinhalte durch eine Umschreibung der Zustände, unter denen der betreffende Inhalt bzw. das Dokument bearbeitet wurden (vgl. [Bars88, Lam⁺94, Pas⁺98]). Dies wird durch psychologische und kognitionswissenschaftliche Erkenntnisse untermauert (vgl. [Tulv83, Kemk88, Eld⁺94, Srin97, Racc97]). Daher wird von mehreren Forschern versucht, durch eine Indexierung von zu recherchierenden Informationsmengen mit Kontextinformationen, d. h. die Verwendung von Kontext als Zugriffsschlüssel auf Informationen, ein für menschliche Nutzer ergonomisches, da auf die Realweltumstände bezogenes assoziatives Retrieval zu ermöglichen (vgl. [Lam⁺94, Pas⁺98]).

Ein Beispiel stellt die von LAMMING bereits in den frühen 1990er Jahren implementierte *Gedächtnisprothese* dar, welche zu speichernde Informationen, etwa verwendete Dokumente, in einem *episodischen*, d. h. über zufällig zur selben Zeit stattfindende Kontextvorgänge adressierten Speicher ablegt. Zur Beschreibung der jeweiligen Kontextereignisse werden Kontextinformationen verschiedener Kontextdimensionen verwendet. Dazu gehören der *Ort* des Nutzers, weitere im Zimmer befindliche *Personen*, die *Tätigkeiten*, die der Nutzer gerade an seinem PC durchführt sowie die Art und der Zeitpunkt, wann *Nachrichten*, insbesondere elektronische Dokumente und Telefonanrufe, mit anderen Personen ausgetauscht werden. Basierend auf diesen Informationen wird ein kontextindiziertes Retrieval ermöglicht, etwa die „EMail, die ich letzte Woche nach einem Telefongespräch mit Herrn Schmidt an ihn gesendet habe“ zu finden. Dabei können nicht nur ganze Dokumente, sondern auch einzelne Teile davon identifiziert werden. KLEMKE verwendet zur Anreicherung des Dokumentations- und Retrievalprozesses ebenfalls Kontextinformationen, um für die aktuelle Situation des Recherchierenden passendere Such-

ergebnisse liefern zu können (vgl. [Klem99]). DOBSON klassifiziert aufbauend auf der Annahme, dass ein Nutzer verschiedene Aufgaben nicht zeitgleich ausführt gespeicherte Informationen (Notizen) automatisch durch Assoziation mit den aktuell vom Nutzer bearbeiteten Prozessen, wodurch ebenfalls ein assoziatives, kontextbezogenes Retrieval etwa aller zu einem Prozess gehörenden Notizen und Dokumente ermöglicht wird (vgl. [Dobs03]).

Bezüglich des Themas dieser Arbeit kann festgestellt werden, dass viele kontextbezogene Retrievalsysteme, u. a. das von LAMMING implementierte System, direkt auf einem eigenen Kontexterfassungssystem aufsetzen und deshalb nicht auf weitere Kontextinformationen und -dimensionen erweitert werden können (vgl. [LaNe92, Lam⁺94]). Dem gegenüber ermöglicht das in dieser Arbeit entwickelte umfassende Kontextmodell nicht nur wesentlich flexiblere und leistungsfähigere, sondern auch ökonomischere Retrievalvorgänge, da sie nicht nur sämtliche zur Konfiguration des Retrievalvorganges notwendigen Informationen enthalten, sondern auch selbst als Informationsbasis über Realweltobjekte fungieren können. Dabei können sowohl betriebliche als auch externe Informationsmengen miteinander integriert und dadurch die betriebliche Informations- und Wissensbasis verbreitert werden.

3.4.1.4 Persönliches Informationsmanagement

Informationsmanagement spielt sich sowohl auf einer persönlichen als auch auf einer organisatorischen Ebene ab. NASTANSKI betont die Bedeutung des persönlichen Informationsmanagements (engl. *personal information management*), worunter die Organisation der Informationen, die am persönlichen Arbeitsplatz eines Nutzers anfallen, verstanden wird (vgl. u. a. [Nast95, S. 274]). Dazu gehört die selbsttätige Erkennung und Speicherung von alltäglichen Informationsarten, unter anderem EMails, To-Do-Listen, Linklisten, Notizen, Logins, Daten oder Termine. Eine informationstechnische Unterstützung kann durch Termin- und Aktivitätenplaner, Dokumentverwaltungssysteme, Know-How-Manager und andere Komponenten von Groupware-Systemen erfolgen, wodurch das dadurch gespeicherte persönliche Wissen auch gleich kollektiviert, d. h. weiteren Personen zugänglich gemacht werden kann (vgl. u. a. [Nast89, Nast90, Nast92, Nast95]).

Ein persönlicher Informationsmanager entspricht also einem persönlichen „Sekretär“, der dem Nutzer während seiner Tätigkeiten informationsbezogen zuarbeitet, d. h. den Nutzer von der Organisation seiner persönlichen Informationen im laufenden Betrieb entlastet, damit sich dieser voll auf die inhaltlichen Aspekte seiner Arbeit konzentrieren und diese flüssig – d. h. ohne sie für explizite Informationsverwaltungsaufgaben zu unterbrechen – bearbeiten kann. Im Gegensatz zur manuellen Verarbeitung ermöglicht die automatisierte Erfassung, Klassifizierung, Speicherung und Organisation der zu verwaltenden Informationen eine verbesserte Verfügbarkeit von Informationen sowohl hinsichtlich der erfassten Menge, Detailierung als auch Indexierung für ein späteres Retrieval.

Durch einen kontextsensitiven elektronischen Informationsmanager, der z. B. mittels mit in die Kleidung integrierten tragbaren Sensoren arbeitet, können nun auch spontan in der Realwelt auftauchende Informationen erfasst werden (vgl. [Sta⁺98], Kap. 3.3.1 („Wearable Computing“, S. 54)). Dabei werden an den kontextsensitiven Assistenten gerichtete, mittels tragbarer Kameras oder Mikrofonen erfasste Nutzergesten zur Markierung der aktuellen Situation verwendet, etwa durch Zeigen auf Objekte wie etwa Schriftstücke, die daraufhin abfotografiert und im Informationsbestand des Nutzers archiviert werden

(vgl. [Sta⁺00, Ash⁺01, Star02b, Sta⁺03]). Die Indexierung der aufgezeichneten Inhalte mit aktuellen Kontextdaten erlaubt das bereits vorgestellte kontextbezogene Retrieval zu späterer Zeit.

Dadurch kann der Nutzer bei alltäglichen Informationsmanagement- und Retrievaltätigkeiten unterstützt werden, etwa durch Anbieten von weiterführenden Informationen und Aktionen zu bestimmten Realweltobjekten und Informationen.

Beispiel 8 *Zu visuell erkannten Telefonnummern in der Realwelt können bei Bedarf mittels Wearable Computing der entsprechende Name der Zielperson sowie ein Kontrollelement, um diese Nummer jetzt anzurufen, eingeblendet werden. Wenn ein Telefon angeblickt wird, kann ein elektronisches Telefonverzeichnis eingeblendet werden (vgl. [Sch⁺01a]).*

3.4.1.5 Kontextsensitive Verzeichnisse

Sogenannte *kontextsensitive Verzeichnisse* enthalten keine statischen Informationen wie z. B. eine auf Papier gedruckte Telefonliste, sondern die jeweils aktuell gültigen Werte, d. h. im Fall der Telefonliste die Telefonnummer, unter welcher die dargestellten Nutzer gerade erreichbar sind sowie die Information ob sie gerade angerufen werden können/möchten oder nicht, was z. B. durch deren augenblickliche Situation (in einem Meeting) oder den Zustand ihres Mobiltelefons signalisiert wird (lautlos geschaltet bedeutet keine Anrufe erwünscht) (vgl. [Sch⁺01b]).

Kontextsensitive Verzeichnisse benötigen also durchaus heterogene Kontextinformationen, um brauchbare Ergebnisse anzeigen zu können. Die Erstellung eines umfassenden, aus diversen Kontextinformationen integrierten Realweltmodells ermöglicht im Gegensatz zu den existierenden Ansätzen die Realisierung solcher kontextsensitiver Verzeichnisse praktisch als „Abfallprodukt“, d. h. ohne zusätzliche Kosten (vgl. [Sch⁺01b]).

3.4.1.6 Kontextsensitive Informationsdarstellung

Virtual Information Towers, also virtuelle Litfaßsäulen, definieren ein an einen Ort gebundenes Informationsangebot mit begrenzter Sichtweite sowie Ausstrahlungswinkel (vgl. [Leo⁺99]). REKIMOTO geht mit der Metapher elektronischer Notizzettel, die nur unter bestimmten Bedingungen, etwa für bestimmte Personen, sichtbar werden, einen Schritt weiter (vgl. [Rek⁺98, S. 75]). Einen ähnlichen Ansatz verfolgt SCHMIDT mit einer elektronischen Wandzeitung, die außer allgemeinen Informationen speziell für davor befindliche Nutzer bestimmte Informationen – etwa deren Terminplan – anzeigt (vgl. [ScGe01, S. 9 f]).

Genau wie bei kontextsensitiven Verzeichnissen ermöglicht die Verfügbarkeit eines globalen Kontextmodells, welches diverse in diesem Zusammenhang benötigte Informationen integriert, die sehr einfache und kosteneffektive Realisierung der an dieser Stelle besprochenen adaptiven Informationsdarstellungsmethoden. Erwähnt sei an dieser Stelle fernerhin, dass die jeweilige Darstellungsanwendung durch den Ansatz, Kontextinformationen systematisch zu integrieren und durch Zugriffsschutzmechanismen zu beschreiben, in einer kontrollierten Weise auf Informationsbestände zugreifen kann, die ohne einen solchen Ansatz nicht direkt für eine solche Anwendung verfügbar wären.

3.4.1.7 Kontextbezogene Informationsorganisation

Bei der Kontextbezogenen Informationsadressierung werden normale Gegenstände aus dem Realweltbereich als Referenz für elektronisch gespeicherte Informationen verwendet. Motivation hierfür ist der Versuch, die besonderen Fähigkeiten des Menschen zur Verarbeitung und Organisation der physikalischen Realität, welche seinen Lebensraum darstellt und auf die er folglich „optimiert“ ist, zur Organisation elektronischer Informationsmengen zu verwenden.

Daher werden beliebige Realweltobjekte, meistens kleine, durch kontextsensitive Informationssysteme leicht erfassbare Gegenstände oder gleich die Personen selbst als Ikonen (engl. *icons*) für ganze Informationssammlungen verwendet (vgl. [BaKi01]). Ein Nutzer, genauer dessen Gesicht oder Fingerabdruck, kann z. B. als Pointer auf dessen persönliche Dokumente verwendet werden, wodurch diese Dokumente an all den Geräten (und nur dort) verfügbar sind, an denen sich der Nutzer befindet und identifiziert. Andere Ansätze unterteilen die physikalische Welt in verschieden große Quader, denen Informationen zugeordnet werden (vgl. [Fitz93, Spoh99, Pasc97]).

Zugegriffen werden diese Informationen über ortsbezogene Endgeräte, welche Informationen zu den in der Nähe befindlichen Objekten anzeigen (vgl. [Abo⁺97, Vol⁺99, Dav⁺01]) oder *Augmented Reality*, welche die betreffenden Informationen direkt in das Sichtfeld des Nutzers einblendet (vgl. Kap. 3.3.2 („Augmented Reality“, S. 55)). Da hierdurch beliebige Informationen organisiert werden, müssen diese vorher entsprechend integriert und den betreffenden Realweltobjekten zugeordnet werden.

3.4.2 Entscheidungsunterstützung

Information und Wissen stellen die Basis für Entscheidungen in allen Phasen des ökonomischen Entscheidungsprozesses, d. h. sowohl bei der Problemidentifikation, der Ermittlung alternativer Handlungen, der Auswahl der am besten geeigneten Alternative sowie der Kontrolle der Zielerreichung dar. Für jeden dieser Schritte werden zwei Arten von Wissen benötigt: Über die vorliegende Situation und wie in dieser Situation zu entscheiden ist. Dieses Wissen wird auch *Kontextwissen* genannt (vgl. [PoBr99, PoBr01]). Da sich ökonomische Entscheidungsprozesse im allgemeinen auf Probleme in der Realwelt beziehen, spielen realweltbezogene Informationen dabei eine tragende Rolle. Insbesondere interessieren dabei Informationen, welche Realweltobjekte auf welche Art in die jeweilige Problemstellung involviert sind, welche Entscheidungsparameter welche Werte aufweisen, was diese Informationen besagen und an welchen Stellgrößen man etwas verändern kann, um unterschiedliche Ergebnisse zu erzielen. Zur Entscheidungsunterstützung wurden eine Reihe unterschiedlicher technischer Maßnahmen realisiert, welche im folgenden separat hinsichtlich ihres Bedarfes nach umfassenden Kontextinformationen untersucht werden.

Entscheidungsprozesse werden maschinell durch *Decision Support Systeme* (DSS) unterstützt. Basis der Entscheidungsfindung stellen betriebsinterne Informationen aus den PPS-Systemen dar. Diese können jedoch durch realweltbezogene Informationen ergänzt und der gesamte Entscheidungsprozess damit näher an der Realwelt und damit an den konkret gegebenen Rahmenbedingungen, unter denen die jeweilige Entscheidung zu fällen ist, orientiert werden.

Beispiel 9 Zur Planung von Fahrtrouten müssen nicht nur die durchschnittlichen Fahrzeiten der Transportfahrzeuge, sondern auch die Häufigkeit von Verkehrsstörungen auf bestimmten Straßen sowie ggf. die zu erwartenden Wetterverhältnisse auf den jeweiligen Fahrtrouten berücksichtigt werden.

Umfassend integrierte Kontextinformationen stellen also eine Informationsbasis dar, die realweltbezogene Parameter für einen Decision-Support-Prozess bereitstellen kann.

Ebenfalls spielen bei der Entscheidungsunterstützung Verfahren des maschinellen Lernens, insbesondere das *Erfahrungsmanagement* eine Rolle, welches Entscheidungen von menschlichen Experten in konkreten Problemstellungen aufzeichnet und auf ähnliche Problemstellungen anzuwenden hilft. Detaillierter wird darauf jedoch aufgrund der bedeutenderen Relevanz von Erfahrungsmanagement hinsichtlich des Wissensmanagements erst im folgenden Unterkapitel eingegangen.

3.4.3 Wissensmanagement

In der postindustriellen Gesellschaft ist Wissen ein relevanter ökonomischer Faktor geworden. Der Marktwert¹¹ wissensintensiver Unternehmen kann ihren Buchwert¹² um ein Vielfaches übersteigen. So war der Marktwert der Firma ORACLE im Jahr 1998 etwa 14 mal höher als ihr Buchwert (vgl. [Svei98]). Die Differenz stellt den Wert der *immateriellen Vermögenswerte*, u. a. des im Unternehmen existierenden Wissens dar (vgl. [Svei98, Fors99]). Daher stellt der Umgang, d. h. die Erfassung, Speicherung und Verbreitung von Wissen und Wissensträgern, eine zentrale Herausforderung für fast jedes Unternehmen dar und sollte entsprechend systematisch und formalisiert gehandhabt werden.

Unter *Wissen* werden die Kenntnisse und Fähigkeiten, die Individuen oder Organisationen zur Lösung von Problemen einsetzen, verstanden. Dies umfaßt sowohl theoretische Erkenntnisse als auch praktische Alltagsregeln und Handlungsanweisungen (vgl. [PiFr88]). Wissen entsteht, indem die (semantisch aussagefähigen) Informationen mit pragmatischer Bedeutung versehen werden, also in Zusammenhänge mit anderen Informationen eingeordnet werden und so als Grundlage für konkret zu treffende Entscheidungen fungieren. Dabei müssen verschiedene Informationen über die Realwelt (Kontextinformationen) mit Problemlösungsinformationen verknüpft werden, so dass am Ende Schlußfolgerungen über auszuführende Handlungen zur Lösung der gegebenen Problemstellung möglich werden. Der Prozess der Wissensbildung besteht also letztlich aus der Kombination von Kontextinformationen mit entscheidungsorientierten Informationen. Kontextinformationen und -modelle spielen daher eine zentrale Rolle im Wissensmanagement (vgl. [Dave96]).

Zu den Verfahren des Wissensmanagements zählen explizite Beschreibungen und Festlegungen, wie Wissen zur betrieblichen Wissensbasis hinzugefügt, bewertet, weitergegeben und gelöscht wird. Die betriebliche Wissensbasis wird dabei als sozio-technisches System, d. h. aus menschlichen und maschinellen Wissens- und Kompetenzträgern bestehend verstanden (vgl. [Hei⁺96, Schw01, ScGe03]). Entsprechend gibt es zwei sich ergänzende Ansätze des Wissensmanagements auf der technischen bzw. der organisatorischen Ebene, die das explizite bzw. implizite Wissen durch entsprechende technische bzw. orga-

¹¹Der Marktwert eines Unternehmens entspricht seinem Börsenwert, also dem Wert, den der Markt (die Börse) einem Unternehmen zumisst, d. h. der Gesamtwert der Aktien des Unternehmens

¹²Der Buchwert entspricht dem offiziellen Eigenkapitalwert des Unternehmens, also seinen materiellen Vermögenswerten.

nisatorische Regelungen und Methodologien handzuhaben versuchen (vgl. [NoTa95, Fors99, Mino01]). Beide Ansätze werden im folgenden getrennt untersucht.

3.4.3.1 Organisationale Perspektive

Der in Form von organisatorischen Regelungen, z. B. durch gezielte Teambildung von Wissensträgern und Wissensempfängern, ermöglichte zwischenmenschliche Austausch von impliziten Fähigkeiten und Kenntnissen verschiedener Wissensträger im Unternehmen kann u. a. durch automatisiert generierte Kompetenzprofile von Mitarbeitern profitieren, um Träger des gesuchten Wissens im Unternehmen zu identifizieren. Dazu sind nutzerbezogene Kontextinformationen auszuwerten, die üblicherweise in Form von Nutzerprofilen dargestellt werden und durch die in Kap. 3.4.1.1 („Informationsleitstand“, S. 58) beschriebenen Verfahren bestimmt werden können. Dieses auf umfassenden Kontextinformationen über die Wissensträger basierende Wissen kann zur Optimierung der organisatorischen Wissensmanagemententscheidungen verwendet werden, d. h. ermöglicht die gezielte Bildung von Arbeitsgruppen, um den dadurch entstehenden Wissensaustausch zu maximieren.

3.4.3.2 Technische Perspektive

Wissen hat durch seinen handlungsentscheidenden, d. h. tätigkeitsorientierten Charakter schon implizit einen Kontextbezug, da sich Wissen mittel- oder unmittelbar auf Tätigkeiten in der Realwelt bezieht. Die Nutzung automatisierter Verfahren zur Erfassung kontextbezogener Informationen erschließt folglich ebenfalls Automatisierungspotenziale im technischen, d. h. durch technische Mittel realisierten Wissensmanagement. Motivation hierfür ist das Bestreben, wertvolle Informationen für das Unternehmen durch Explizierung und Kollektivierung von individuellem Wissen verfügbar zu machen, denn Wissen ist an individuelle menschliche Aufgabenträger gebunden und stellt daher genau wie diese eine vergängliche Unternehmensressource dar, die mit dem Verlassen des jeweiligen Mitarbeiters für das Unternehmen nicht mehr nutzbar ist. Ziel des technischen Wissensmanagements ist daher, möglichst viel Wissen in elektronischer Form zu explizieren, um es allen Mitarbeitern und betrieblichen Anwendungssystemen des Unternehmens zugänglich zu machen (vgl. [NoTa95, Mino01, OsWa97]). Im Folgenden werden verschiedene technische Wissensmanagementkonzepte und deren Bezug zu umfassenden Kontextinformationen dargestellt.

3.4.3.3 Organisational Memory Systeme

Organisational Memory Systeme (OM-Systeme) als zentrale Komponenten einer technischen Wissensmanagementstrategie erfassen, speichern und verbreiten organisations- und unternehmensbezogenes Wissen in einer expliziten, persistenten Form, z. B. als kontextbezogen indizierte Dokumentsammlungen, Best-Practice-Reports, Emails, Diskussionsforen sowie Referenzen auf Mitarbeiter oder archivierte Workflow-Instanzen, die sich mit den der aktuellen Situation vergleichbaren Problemstellungen beschäftigt haben bzw. dafür zuständig sind. Anwendung finden solche Systeme im Rahmen eines z. T. auch proaktiven *Information Supports* von Individuen hinsichtlich Organisationswissen, wodurch besonders wissensintensive Arbeitsprozesse effektiver und effizienter gestaltet werden sollen (vgl. Kap. 3.4.1 („Informati-

onsmanagement, Information Support“, S. 58)). Diesem Bereich sind auch *Group Memory*-Systeme aus dem CSCW-Bereich zuzuordnen (vgl. [Choo95, Hei⁺96, KuAb97, Har⁺97, Abe⁺98, Reim98, Klem99, Klem00, Maus01]).

KLEMKE erweitert OM-Systeme konzeptuell um eine kontextbezogene Inhaltsindizierung (vgl. [Klem99, S. 483]). Dabei werden vor allem diejenigen Kontextdimensionen verwendet, die ein Individuum in einer Organisation wahrnimmt, also die Rollen und Positionen der beteiligten Personen aus einer Mitarbeiterdatenbank, gegenwärtige Problembereiche und Lösungsschritte aus einem Workflowsystem sowie weitere Deskriptoren über die aktuelle Situation. Verwendung finden können solche kontextbezogen indizierten OM-Systeme neben den bereits angesprochenen effektiveren Retrievalmöglichkeiten u. a. proaktiven im *Information Support* für betriebliche Angestellte, wobei ein Kontextagent die Situation der Angestellten überwacht und bei einem Kontextwechsel proaktiv die für diese Situation relevanten Dokumente beschafft (vgl. [Klem99, Klem00]).

3.4.3.4 Erfahrungsmanagement

Zu den technischen Wissensmanagementverfahren zählt auch die Erkennung und Speicherung von beispielhaftem Problemlösungswissen anhand der Protokollierung der Tätigkeiten von menschlichen Domänenexperten durch *Erfahrungsmanagementsysteme* (engl. *Experience Management Systems*). Dieser letztlich für die Leistungserstellung relevante Teil des Wissens wird auch als *Know How* bezeichnet (vgl. [PrKn98]).

Dabei werden typische Problemszenarien zusammen mit anzuwendenden Problemlösungsschritten mittels Deskriptoren beschrieben und in einer Wissensbank abgelegt. Dadurch können sie bei Bedarf auf ähnliche, neu auftretende Problemstellungen angewendet werden, indem die aktuelle Situation ebenfalls durch Deskriptoren beschrieben wird, in der Wissensbank nach den ähnlichsten gespeicherten Situationen gesucht und die damit assoziierten Problemlösungsschritte auf die aktuelle Situation adaptiert angewendet werden (vgl. [Berg01, Schn02, Berg02]). Ebenfalls möglich ist der Einsatz von *Knowledge Discovery* genannten *Data Mining*-Verfahren zur (halb)automatisierten Identifikation des verwendeten Lösungswissens mittels Recherche in den aufgezeichneten Problemlösungsschritten.

Eins der größten Hindernisse stellt in diesem Zusammenhang die aufwändige und teure Modellierung einer ausreichenden Zahl von Beispielszenarien durch menschliche Experten dar, was eine notwendige Voraussetzung für ein erfolgreiches Erfahrungsmanagement darstellt. Durch automatisiert bereitgestellte umfassende Kontextinformationen wäre es möglich, diese Szenarien durch Beobachtung eines menschlichen Experten im laufenden Betrieb automatisch zu generieren und Entscheidungsunterstützungssysteme dadurch nahezu selbsttätig zu trainieren, ggf. unter Einsatz der bereits erwähnten *Knowledge Discovery*-Verfahren. Dadurch können Erfahrungsmanagementsysteme nicht nur in Sonderfällen, sondern in wesentlich breiterem Umfang eingesetzt und so eine effektivere Explikation von implizitem Expertenwissen betrieben werden.

Dabei muß das Verhalten des menschlichen Experten in möglichst vielen Kontextdimensionen gleichzeitig beobachtet werden, um möglichst viele treffende Problemdeskriptoren und -lösungsschritte erfassen zu können. Dies ist nur mittels eines umfassenden Kontextmodells, welches diverse heterogene Kontextinformationen integriert, notwendig.

3.4.3.5 Fazit

Schon vor geraumer Zeit wurde relativ einhellig ein Bedarf nach Wissensmanagement festgestellt. Das INSTITUT FÜR LERNENDE ORGANISATION UND INNOVATION stellte durch Umfragen fest, dass die überwiegende Anzahl der Unternehmen (87%) dem Wissensmanagement eine hohe bis sehr hohe Bedeutung beimessen. PALASS geht sogar von 96% der Unternehmen aus (vgl. [O.V.97, Pala97]). Trotzdem sind entsprechende praktische Lösungen nur teilweise umgesetzt worden, denn Wissensmanagement in der einen oder anderen Form wird von lediglich 28% der Unternehmen verwendet. Entsprechend werden auch nur 20-40% des internen Wissens in Unternehmen genutzt (vgl. [ZuSc94, Grou97, Pala97, Nort98]). Dies liegt nicht zuletzt daran, dass solche Systeme ohne eine umfassende Versorgung mit Kontextinformationen wie bereits angedeutet nicht im benötigten Maß automatisiert arbeiten können. Problematisch ist die Bereitstellung einer hinreichend großen Menge von Trainingsdaten, die ohne kontextsensitive Verfahren manuell und damit teuer bereitgestellt werden müssten. Eine Umfrage des Fraunhofer-Instituts für Arbeitswissenschaft hat bereits 1997 festgestellt, dass diejenigen Mitarbeiter, die etwas wissen, keine Zeit haben, ihr Wissen für das Wissensmanagementsystem zu explizieren, da das Verhältnis von aufgewendeter Zeit (für eine manuelle Explikation des Wissens) und erwartetem Nutzen der Wissensexplikation für den betreffenden Wissensträger nicht angemessen korrelieren.

Ein weiteres Problem stellt die Erfassung der Reaktionen des Anwenders auf die vom Wissensmanagement bereitgestellten Hinweise dar, die als *Teachersignal* zur weiteren Anpassung und Verfeinerung der Problemlösungsbasis benötigt werden. Dieses explizite Feedback muß in der Praxis von einem gerade in einer Problemsituation steckenden und damit wahrscheinlich gestressten Nutzer abgefragt werden, was den Nutzer in dieser Situation zusätzlich belastet. Sämtliche genannten Aspekte unterstreichen den Bedarf nach einer kontextbezogenen Unterstützung des Prozesses der Wissenserfassung und -verwertung.

Eine in Form von Context Maps bereitgestellte integrierte Kontext- und Situationsdatenbank würde ein automatisiertes Wissensmanagement ermöglichen, indem durch Beobachtung des Verhaltens des Nutzers in verschiedenen Kontextdimensionen und Anwendungssystemen eine ausreichende Anzahl von Trainingsdaten und Nutzerfeedback bereitgestellt würde. Dieser technologische Ansatz ist mit organisatorischen Maßnahmen, z. B. personalwirtschaftlichen Anreizsystemen zur Erschaffung einer kooperationsfreundlichen Atmosphäre sowie den bereits angesprochenen organisationalen Wissensmanagementverfahren zu kombinieren, um die Bereitschaft der Angestellten zum Austausch von Wissen zu verstärken, da die kontextbezogenen technologischen Verfahren ansonsten auch abgelehnt werden können und damit nicht wirksam werden.

3.4.4 Dokumentation/Logging

Unter *Dokumentation* wird sowohl der Prozess des Sammelns und Speicherns von Informationen aller Art zur Beschreibung von Dingen (Objekten, Vorgängen, Situationen) als auch die durch diesen Dokumentationsprozess entstehende Dokumentenmenge verstanden. Beim Dokumentationsvorgang werden Informationen formalisiert erfasst, durch Metainformationen beschrieben und derartig erschlossen recherchierbar in einer Informationsbasis abgelegt. Dokumentation ist ein wichtiges Instrument zur Planung, Kontrolle, Koordination und Bewertung von betrieblichen Prozessen (vgl. [Henz92, Bud⁺91, Curt01]).

Beim Dokumentationsvorgang betrieblicher Prozesse werden die konkret ausgeführten Prozess-Schritte beschrieben. Dabei können die aus Workflowmanagementsystemen stammenden Prozessinformationen mit Kontextinformationen kombiniert werden. Diese beinhalten nicht nur realweltbezogene Auswirkungen der elektronisch durchgeführten Prozessschritte, sondern beschreiben ebenfalls ausschließlich in der Realwelt durchgeführte Prozessschritte, die nicht durch Workflowsysteme erkannt werden, etwa Telefonanrufe (vgl. [Klem99, Maus01]).

Dies ergibt eine ganze Reihe von Vorteilen. Neben einer kontextbezogenen Indexierung der aufgetragenen Informationen, die zu flexibleren Suchstrategien führt (vgl. Kap. 3.4.1.3 („Information Retrieval“, S. 59)) wird eine verbesserte Nachvollziehbarkeit von Prozessen erreicht, was sich positiv auf deren Abrechenbarkeit bzw. das Qualitätsmanagement auswirkt. Auch können Fehler gezielter eingegrenzt bzw. identifiziert werden.

3.4.5 E-Learning, Computer aided Instruction

Unter *E-Learning*, auch als *Computer-aided-Instruction* bezeichnet, wird die Unterstützung des Lernprozesses menschlicher Personen durch elektronische Medien verstanden (vgl. [Bac⁺02]). Neben einer relativen Unabhängigkeit aller Beteiligten (Lehrer und Lernende) von Ort und Zeit ermöglicht E-Learning vor allem eine Personalisierung von Schulungsmaterial, da dieses i. d. R. nicht statisch auf Papier, sondern mittels adaptierbarer elektronischer Medien bereitgestellt wird.

Während für lediglich *elektronifizierte* Schulungsvorgänge Informationen über den jeweiligen Nutzer und seinen Kenntnisstand im zu vermittelnden Fachgebiet als Basis für grundlegende Adaptionen der Lerninhalte benötigt werden, geht der Kontextinformationsbedarf bei fortgeschritteneren situationsbezogenen Lernmethoden, etwa bei einem durch E-Learning unterstützten *Training on the Job* weiter (vgl. [Fei⁺93]). Hier werden im laufenden Betrieb an die jeweilige Situation angepasste Lerninhalte bereitgestellt, wofür entsprechende Informationen aus den folgenden Bereichen benötigt werden.

- Die Identität des Nutzers, welche von kontextsensitiven Identifikationssystemen wie tragbare elektronische Badges oder per Gesichtserkennung bereitgestellt wird (vgl. u. a. [Wan⁺92]),
- Der Kenntnisstand des Nutzers hinsichtlich der Lerninhalte, welcher in Form von Wissensprofilen im E-Learningsystem bereitgestellt wird. Dieses Profil enthält Informationen darüber, welche Lerneinheiten der Nutzer bereits erfolgreich durchlaufen und angewendet hat und ggf. der Wissensstand des Nutzers in typischen Wissensgebieten.
- Die Rolle und Charakteristika des Nutzers. Darunter fallen seine Lernziele, d. h. welches Wissen in welcher Detailierungsstufe benötigt wird. Diese Informationen können aus seiner Rolle im Unternehmen abgeleitet werden, die aus CSCW-Systemen stammen kann.
- In der aktuellen Situation relevante Probleme oder Aufgaben, zu welchen Informationen angeboten werden sollen. Diese Informationen werden typischerweise von CSCW- oder Erfahrungsmanagementsystemen erfasst.
- Personen, die Kenntnisse in den betreffenden Gebieten haben. Diese können als Ansprechpartner fungieren, falls eine eigenständige Lösung der gestellten Aufgabe trotz E-Learning nicht möglich oder nicht gewünscht ist (vgl. [HeSi01]).

Diese Kontextinformationen stammen wie dargestellt aus sehr unterschiedlichen Kontextquellen und müssen in geeigneter Weise zusammengetragen und miteinander kombiniert werden, um eine Entscheidung über die zu vermittelnden Lerneinheiten zu ermöglichen.

Ein Beispiel für ein kontextbezogenes Computer-Aided-Instruction sei in Form der Wartung von technischen Anlagen gegeben. Problematisch dabei ist der Zugriff auf benötigte Informationen aus der Dokumentation während des Wartungsvorganges. Dokumentationen werden traditionell als Handbuch in gedruckter Form bereitgestellt, wodurch es nötig ist, die aktuelle Wartungstätigkeit zu unterbrechen und eventuell in der Hand befindliche Werkzeuge beiseite zu legen, um das jeweilige Buch mit ggf. dicken Arbeitshandschuhen auf der Suche nach benötigten Detailinformationen durchzublättern. Dies manchmal aus Platzgründen auch gar nicht möglich sein kann, etwa bei Wartungsarbeiten an unzugänglichen Teilen im Innern einer Hubschrauberturbine. Der erfolgreiche Einsatz von *Wearable Computing* in diesem Bereich offenbart einen möglichen Ausweg (vgl. Kap. 3.3 („Kontextbezogene betriebliche Anwendungssysteme“, S. 53)). Informationen können dabei in jeder Situation unter Einsatz nur einer Hand ein- und ausgegeben werden (vgl. [CaMi92, O.V.04g]). In Verbindung mit kontexterfassenden Systemen kann dieser Informationszugriff nochmals rationalisiert werden, indem eine Vorauswahl der benötigten Informationen getroffen und bei Bedarf in das Sichtfeld des Nutzers eingeblendet wird (siehe Abb. 24 auf Seite 56).

3.4.6 Administration von Anwendungssystemen

Die komplexen, zahlreichen und dynamischen Computersysteme in modernen Szenarien, u. a. im *Pervasive Computing*, stellen neue Herausforderungen an die Administration, d. h. deren Installation und Wartung, dar. Während die heute übliche Wartung durch Fachkräfte aufgrund der Vielzahl der vorhandenen Geräte ökonomisch nur selten sinnvoll sein wird, kann von einem typischen Endnutzer weder die benötigte Qualifikation noch die Bereitschaft zur Administration dieses Geräteparks erwartet werden. Pervasive Computersysteme müssen sich daher in der einen oder anderen Form selbst administrieren und konfigurieren. Dazu werden systematische Lösungen für die Administration komplexer Infrastrukturen benötigt (vgl. [Pesc02]).

Unter *Konfigurationsmanagement* (engl. *configuration management*) wird der systematische Ansatz zum Erkennen, Archivieren und Recherchieren der Konfiguration von Anwendungssystemen verstanden. Unter *Konfiguration* wird in diesem Fall der gesamte Aufbau eines Anwendungssystems aus verschiedenen Modulen, die Einstellungen der einzelnen Module sowie die Verknüpfungen der Module untereinander verstanden. Ein Konfigurationsmanagementsystem erlaubt das selbsttätige Erkennen der Konfiguration von technischen Systemen zu beliebigen diskreten Zeitpunkten, die Archivierung sowie das Wiederherstellen derselben bzw. die Übertragung der Konfigurationseinstellungen auf ähnliche Systeme und stellt somit einen systematischen Ansatz zur Administration aller Arten von Anwendungssystemen dar. Durch den Einsatz von Konfigurationsmanagementsystemen können die für den Betrieb von sehr komplexen Anwendungssystemen notwendigen Installations- und Wartungskosten durch Verringerung zumindest von redundantem, d. h. mehrfach durchzuführendem menschlichen Administrationsaufwandes minimiert werden. Dies bezieht sich insbesondere auf die Vereinheitlichung und Verringerung von redundantem Konfigurationsaufwand durch Übertragung funktionierender Konfigurationen auf ähn-

Konfigurationsdimension	Beschreibung	Beispiel
<i>nutzerspezifisch</i>	Einstellungen für einen Nutzer	Mindest-Schriftgröße, Zahlungsmodalitäten und -präferenzen
<i>anwendungsspezifisch</i>	anwendungsspezifische Einstellungen	Adressen von Anwendungsservern bei Client-Server-Betrieb
<i>gerätespezifisch</i>	Einstellungen für einzelne Geräte	Zugriffsrechte, zu verwendende Netzwerkverbindung
<i>situationsspezifisch</i>	allgemeine situationsbezogene Einstellungen	keine akustische Interaktion im Theater

Tabelle 8: Konfigurationsdimensionen

liche Systeme sowie die Erkennung von Fehlern durch Abgleich der Einstellungen fehlerkonfigurierter Anwendungssysteme mit als funktionierend bekannten Konfigurationen aus dem Konfigurationsarchiv (vgl. [Alla97, SiDe93, Morr91]).

Klassischerweise werden von Konfigurationsmanagementsystemen nur Bausteine von Unternehmensanwendungen berücksichtigt. Die Nutzung von Konfigurationsmanagement beim Pervasive und Ubiquitous Computing macht den Einsatz von Kontexterfassungssystemen für die technische Kontextdimension notwendig (vgl. [Bru⁺00, Pha⁺01, Pon⁺01]). Diese Systeme ermöglichen eine Orientierung im *technischen Kontext*, indem sie die per Netzwerk erreichbaren Geräte und Dienste lokalisieren und identifizieren. Ansätze dazu stellt die OPEN SERVICES GATEWAY INITIATIVE (OSGi, vgl. [O.V.04d, O.V.04c]) bereit. Basierend auf diesen technischen Kontextinformationen kann in Kombination mit den in Tab. 8 dargestellten weiteren Kontextinformationen u. a. über den Benutzer und seine Tätigkeiten eine an die aktuelle Situation angepasste Konfiguration der pervasiven Netzwerkumgebung durchgeführt werden (vgl. [Hod⁺97, Wer⁺01, Jons02]).

3.4.7 Fehlertolerante Systeme und Prozesse

Das Übertragen von kritischen Funktionen auf maschinelle Aufgabenträger verringert auf der einen Seite das Risiko menschlichen Fehlverhaltens, macht das Unternehmen jedoch vom einwandfreien Funktionieren der verwendeten Technik abhängig. Die durch den Ausfall von Technik entstehenden Kosten können sehr schnell enorme Ausmaße annehmen. Die Kosten für den durch Virenbefall hervorgerufenen branchenweiten Ausfall von Computerarbeitsplätzen haben sich in der Praxis beispielsweise schon nach mehreren Stunden auf mehrere Milliarden Dollar summiert (vgl. [Pers03]). Bei Totalausfall von zentralen Komponenten des rechnergestützten Anwendungssystems sind selbst globale Unternehmen nach sehr kurzer Zeit nicht mehr überlebensfähig. Bei Dienstleistungsunternehmen beträgt diese Zeitspanne für den Totalausfall des Rechenzentrums einige Tage, bei Unternehmen im Finanzsektor nur noch 12-24 Stunden (vgl. [Haas96]).

Technische Systeme werden jedoch, da sie von fehlbaren Menschen mittels fehlerhafter Werkzeuge erschaffen werden, prinzipbedingt niemals fehlerfrei funktionieren. In diesem Zusammenhang sei an die fünf- bis sechsstelligen (bereits bekannten!) Fehlerzahlen in modernen Betriebssystemen erinnert (vgl. [Pers00]). Daher stellen Verfahren, um die Funktionalität operativer sowie administrativer

Fehlerart	Realweltsymptom	Handhabung
fehlerhafte Ausgangsstoffe	Erkennung von äußeren Schäden z. B. an Transportpaletten durch Kameras	Aussortieren
fehlerhafte Produktionsanlagen	Erkennung von mechanischen Schäden durch Veränderungen des Betriebsgeräusches von Maschinen, Produktion von Ausschuß trotz einwandfreier Ausgangsstoffe, Statusmeldungen von Monitor-Komponenten	Abschalten der Maschine, Nutzung von Ersatzanlagen, Anforderung von Reparatur
fehlerhafte Konfiguration	Treiberprobleme, Nichtverfügbarkeit von Diensten	Rekonfiguration durch Configuration Management Systeme
fehlerhafte Distribution	Ressourcenengpässe: Komponentenauslastung übersteigt obere oder untere Grenzen	Optimierung der Ressourcendistribution im Produktionsprozess mittels operativer und administrativer Steuerungssysteme

Tabelle 9: Arten und Handhabungen typischer kontextbezogen erkennbarer Fehlerklassen

Produktions- und Anwendungssysteme aufrecht zu erhalten, sowie Strategien für den Umgang mit Fehlern und Ausfällen eine absolut notwendige Komponente bei der Nutzung von technischen Systemen dar.

Unter *Fehlertoleranz* wird die Eigenschaft von Geräten, Systemen oder Prozessen verstanden, das vom Nutzer beabsichtigte Ergebnis trotz erkennbar fehlerhafter Eingabe oder Bedienung zu liefern, wobei der Nutzer keinen oder nur einen minimalen Korrekturbeitrag leistet (in Anlehnung an DIN EN ISO 9241-10, Abschnitt 3.6 ([DIN96])). Das in diesem Zusammenhang selbstständige Beheben von Engpässen, Unregelmäßigkeiten und Fehlern durch maschinelle Prozesse wird als *Autonomic Computing* oder *selbstheilende Systeme* bezeichnet (vgl. [Will96, IBM03, Rush02, WiNa96, KeCh03]). Grundsätzlich werden in diesem Bereich zwei orthogonale, sich ergänzende Ansätze unterschieden, zum einen die Vorhaltung ausreichender Reserve- und Ersatzkapazitäten wichtiger Systemen (Redundanz), zum anderen das fehlertolerante Design von Anwendungssystemarchitekturen. Entsprechend werden je nach Art der auftretenden Unregelmäßigkeit fehlerhafte Komponenten abgeschaltet und ggf. repariert, wobei die ausgefallene Funktionalität von redundanten Komponenten übernommen werden muß, oder im laufenden Betrieb umkonfiguriert, um die neue Situation angemessen handhaben zu können.

Der Kontextbezug bei der Fehlererkennung und -behebung ist offensichtlich. Kontextinformationen bilden Realweltaspekte der betrieblichen Prozesse ab, daher stellen ungeplante Veränderungen von Kontextinformationen Indikatoren zur Erkennung von auftretenden Problemen dar (vgl. [Har⁺03, S. 3]). Zur sicheren Erkennung von Fehlern gehört neben der Auswertung interner *Statusmeldungen* der PPS-Anlagen in vielen Fällen daher auch die Kontrolle realweltbezogener Prozessindikatoren, was durch Kontexterfassungssysteme erfolgt (vgl. Tab. 9). Neben dem Einsatz von entsprechenden *Sensoren* kann die Behebung von Fehlern darüber hinaus den Einsatz von kontextverändernden *Aktoren*, d. h. Software-Agenten oder Realwelt-Robotern, nötig machen, wobei sich in beiden Fällen ein Informations- bzw. Aktionsbedarf in unterschiedlichen Kontextdimensionen ergibt. Die daher notwendige Integration ver-

schiedener Kontexterfassungs- und -adaptionssysteme erfordert eine umfassende Integrationsstrategie für heterogene Kontextinformationen aus verschiedenen Datenquellen, wie sie in dieser Arbeit entwickelt wird.

Dadurch werden die fehlerverursachten Mehrkosten von technischen Systemen im betrieblichen Leistungserstellungsprozess minimiert¹³. Außerdem können durch ein automatisiertes Fehlermanagement die *Total Cost of Ownership* (TCO, vgl. [Scri02]) für technische Systeme verringert werden, wobei sich dies besonders auf die Posten für Support und Wartung, insbesondere auf den Umfang des menschlichen Wartungspersonals bezieht, da teure menschliche Administratoren nicht mehr für bekannte, standardisierte Fehler bemüht werden müssen.

3.4.8 Lagerhaltung

Das Lagergeschäft stellt die bedeutendste Geschäftsart zur mengenmäßigen, zeitlichen und räumlichen Überbrückung von materiellen Produkten dar, welche Aufgabe des Handels ist. Neben der Ein- und Auslagerung sind innerhalb des Funktionsbereiches von Lagern vor allem die Verwaltung der Lagerstruktur, die Lagersteuerung und die Inventurdurchführung zu nennen (vgl. [BeSc04]).

Der Kontextbezug des Lagergeschäftes ist prinzipieller Natur, da vor allem physikalische, also Realweltgüter gelagert werden. Die Verwaltung des Lagerstandes im betrieblichen Umfeld ist in Form von Lagerverwaltungssystemen bereits automatisiert, wird in der Praxis jedoch nur durch Mitzählen der Ein- und Auslagerungsvorgänge realisiert. Daher ist eine periodische Synchronisation des tatsächlichen Lagerstands mit den Inventarlisten notwendig, der als *Inventur* bezeichnet wird. Inventuren werden aufgrund der „Blindheit“ von Lagerverwaltungssystemen hinsichtlich der Realwelt gegenwärtig vorwiegend manuell durchgeführt. Die durch kontextsensitive Technologie ermöglichte Erkennung des tatsächlichen Lagerstandes macht zumindest periodische manuelle Inventuren überflüssig.

Während innerbetriebliche Lagerungen wie bereits dargestellt von spezialisierten Systemen durchgeführt werden, erfordern öffentliche Lagerungsvorgänge wie etwa die möglichst effektive Auslastung eines Parkhauses die Integration verschiedener Kontexterfassungstechnologien, in diesem Fall etwa die Erfassung besetzter Lücken durch das Parkhausssystem sowie die Identifikation und Positionierung einzelner Fahrzeuge durch bordinterne Navigationsgeräte. Auf diesen Aspekt wird in Kap. 6.2.4 („Kontextsensitives Navigationssystem“, S. 186) noch genauer eingegangen.

Ein anderes Anwendungsfeld für kontextbezogene Lagerhaltungskonzepte stellen moderne Controllingkonzepte wie *Lean Production* dar, die auch als *Just-In-Time*-Anlieferung bezeichnet werden. Diese zielen auf die Minimierung von kostenintensiver Zwischenlagerung durch zeitlich exakt aufeinander abgestimmte Produktions- und Logistikprozesse ab, d. h. unternehmensinterne und -externe Distributionsvorgänge von Ausgangsstoffen, Halb- und Fertigproduktion werden mit dem Produktionsablauf sowohl der Zulieferer als auch der Abnehmer koordiniert (vgl. [FeSi98, S. 78], [Wom⁺91], [Ihde01], [Merk95], [Pfoh03]). Während die Disposition von Ressourcen *innerhalb* eines Standortes relativ determiniert ist, unterliegt der Transport von Ressourcen zwischen geografisch entfernten Standorten einem durchaus relevanten Umfang an z. T. unvorhersehbaren externen Störeinflüssen und ist damit nicht vollständig plan-

¹³Fehlerkosten untergliedern sich in interne und externe Fehler- und Fehlerfolgekosten. Weiteres in DIN 55350-11:2004-03, Teil 11 ([DIN04])

bar, sondern muß zur Laufzeit möglichst korrekt in Abhängigkeit der jeweiligen Verhältnisse geschätzt werden. Hinsichtlich des Transportes auf dem Straßennetz werden u. a. detaillierte verkehrsbezogene Kontextinformationen, insbesondere Staumeldungen, Wetterberichte sowie Informationen zu Veranstaltungen, Demonstrationen, Straßensperrungen, Unfälle sowie Katastrophen auf dem geplanten Transportweg benötigt. So kann in Reaktion auf einen unfallverursachten Stau auf der Zubringerstraße eines Produktionsstandortes ein laufender Transport zu diesem Standort zeitnah umgeleitet bzw. entsprechend beschleunigt werden.

Wie aus beiden dargestellten Beispielen ersichtlich ist, müssen die aus unterschiedlichen externen Quellen stammenden Kontextinformationen innerhalb des betreffenden Logistik-Informationssystems in einer möglichst integrierten Form dargestellt werden, um für die Prognose von Transportzeiten und damit die Planung des logistischen Tagesgeschäftes genutzt werden zu können.

3.4.9 Gebäudemanagement

Unter *Gebäudemanagement* wird die Steuerung adaptiver Gebäudekomponenten wie Beleuchtung, Heizung, Belüftung, Kühlung sowie mobiler Gebäudeteile (mobile Trennwände) verstanden, wodurch ein möglichst komfortables und gleichzeitig energieeffizientes Verhalten des Gebäudes in Abhängigkeit von den äußeren Wetter- und inneren Nutzungsverhältnissen erreicht werden soll.

Durch Berücksichtigung von Kontextinformationen kann das Gebäudemanagement hinsichtlich Komfort und Sicherheit optimiert und gleichzeitig kostenreduziert werden (vgl. [Str⁺98, Bru⁺00, Jan⁺01]). In das Gebäude integrierte Sensoren sorgen für detaillierte Kontextinformationen hinsichtlich der aktuellen Nutzung der einzelnen Zimmer, etwa die Anzahl der darin befindlichen Personen, wodurch diese mittels elektronisch steuerbarer Aktoren, z. B. elektronisch steuerbarer Lichtschalter, Klimaanlage oder Jalousien individuell beleuchtet und beheizt werden können. Zu den kostenminimierenden Aspekten zählt das selbsttätige Dimmen der Beleuchtung, sobald sich keine Personen im Zimmer befinden, das automatische Herunterfahren der Heizung bei längerer Abwesenheit (z. B. nachts oder bei Urlaub des Nutzers) sowie das Abschalten der wirkungslos werdenden Air Condition bei geöffnetem Fenster. Ein Beispiel stellt das *Smart Rooms*-Projekt des MIT dar, wobei einzelne Personen und deren Körperhaltung und Gestik interpretiert werden, um z. B. nur die Teile eines Zimmers zu beleuchten, die sich im Blickfeld des Nutzers befinden. Sitzt er etwa vor einem Computer, wird die Beleuchtung des Zimmers hinter ihm gedimmt (vgl. [Pent95, WrSp96, Coe⁺99]). Die durch ein kontextsensitives Gebäudemanagement erzielten Einspareffekte erreichen beim Management ganzer Bürogebäude schnell Hunderttausende bis Millionen Euro, wenn man die jährlichen Betreiberkosten für solche Gebäude zugrunde legt. Ebenfalls zum Gebäudemanagement zu zählen sind Sicherheitsmerkmale wie das automatische Schließen geöffneter Fenster bei Regen oder eine Zentralverriegelung für das Haus, die sämtliche Haustüren verriegelt und die Fenster in eine einbruchssichere Position bringt, sobald der letzte Nutzer das Haus verläßt. Diese tragen gleichzeitig zu mehr Sicherheit und geringeren Versicherungskosten dank minimiertem Diebstahlrisiko in derartig einbruchshemmenden Häusern bei.

Das Gebäudemanagement kann hinsichtlich weiterer Kontextdimensionen erweitert werden. Basierend auf der Identifikation von Nutzern in Kombination mit entsprechenden Nutzerprofilen kann die Raumklimatisierung hinsichtlich der individuellen Präferenzen der jeweiligen Insassen gestaltet wer-

den (vgl. [Hart98]). Eine unerwünschte Folge der nutzungsabhängigen Klimatisierung ist, dass Nutzer jeden Morgen ein nicht beheiztes oder gekühltes Büro vorfinden. Daher müssen die bereits existierenden Gebäudemanagementansätze mit einer Prognose des Nutzerverhaltens kombiniert werden, wodurch sie schon vor ihrer geplanten Benutzung angemessen klimatisiert werden können. Die letztgenannten Bereiche erfordern die Integration heterogener Kontextinformationen, u. a. von Gebäudeinformationen, Nutzerinformationen, die Beobachtung von Nutzergewohnheiten und Terminen verschiedener Art.

3.4.10 Prozessmanagement, Workflow Computing

Die zunehmende Integration hochspezialisierter Akteure in betriebliche Prozesse, verursacht durch die Zerlegung der betrieblichen Prozesskette in immer stärker abgegrenzte Einzelprozesse, verursacht einen immer stärkeren *Koordinationsaufwand*, der sich in einem stetig steigenden Anteil der *Transaktionskosten* am Gesamtumsatz einer Volkswirtschaft niederschlägt. Der Anteil der Transaktionskosten am Bruttosozialprodukt der Vereinigten Staaten stieg im Zeitraum von 1870 bis 1970 von 25 % auf 55 %. Heute dürfte er über 60 % betragen (vgl. Kap. 2.1.2 („Das Unternehmens- und Umweltmodell in der Betriebswirtschaftslehre“, S. 15)).

Zur Unterstützung dieser arbeitsteiligen Geschäftsprozesse in der Realwelt wurden prozessorientierte IT-Verfahren, insbesondere das Workflow-Management, entwickelt. Unter *Workflow Computing* wird eine Form der Telekooperation zur Unterstützung strukturierter arbeitsteiliger Geschäftsprozesse nach festgelegten Regeln und Methoden verstanden. Prozesse werden dabei laut DIN als „*computergestützter, organisierbarer und steuerbarer Geschäftsprozess*“ verstanden und basierend auf Prozessmodellen organisiert und automatisiert (vgl. [O.V.96a, S. 78]). Im Besonderen werden durch Workflowsysteme die einzelnen Prozessschritte gezielt vorbereitet und unterstützt, indem die personellen sowie maschinellen Aufgabenträger entsprechend den zu bearbeitenden Aufgabenstellungen koordiniert sowie die dabei notwendigen Datenflüsse abgestimmt werden (vgl. [Sonn99, S. 83 f], [BeVo96], [Houy96]). Neben den Prozessen und Prozessschritten werden dabei auch beteiligte Personen, deren Aufgaben und Rollen sowie die in den einzelnen Prozessschritten benötigten und produzierten Informationen und Dokumente sowie Informationen über die Zusammenhänge all dieser Objekte modelliert.

Der Kontextbezug von *Workflow Computing* entsteht durch die Orientierung an realweltbezogenen Prozessen. Um diese Prozesse möglichst genau an den tatsächlichen Gegebenheiten zu orientieren, werden aktuelle Umgebungsinformationen benötigt. Zur Koordination von Prozessen werden also neben den in WFMS modellierten prozessinhaltlichen Informationen auch Angaben über prozesseexterne, die jeweilige Situation beschreibende Momente benötigt. Dazu zählen neben Informationen, wann welcher Prozess zu starten ist auch Informationen, um einen einzelnen Prozessschritt abhängig von den verfügbaren Betriebsmitteln oder dem jeweiligen menschlichen Bearbeiter und dessen Kenntnis- und Fähigkeitsstand konkret zu gestalten. Eine detaillierte Kenntnis der aktuellen Situation der Prozessteilnehmer ermöglicht also eine situationsspezifische Prozessgestaltung.

In besonderer Form macht sich dies bei den von HUTH und NASTANSKY beschriebenen *ad-hoc-Workflows* bemerkbar, welche stark strukturierte, jedoch i. A. von den aktuellen Gegebenheiten dynamisch zusammengesetzte und konfigurierte Workflows darstellen. Besonders in diesem Zusammenhang wird immer wieder hervorgehoben, dass heutige Workflowsysteme einen ungenügenden Kontextbezug

aufweisen, d. h. kein umfassendes Konzept beinhalten, um Workflow-Kontext darzustellen und Kontext aus einem Workflow heraus zuzugreifen (vgl. [ReLe00, Maus01]). Fernerhin stellen heutige Workflowsysteme eher technische, aber keine semantischen Informationen zur Verfügung, die zur Interpretation und Voraussage von Nutzerverhalten führen könnten. JABLONSKI und GOESMANN beschreiben daher Erweiterungen der Struktur und Schnittstellen von WFMS, um diesen Anforderungen besser gerecht zu werden (vgl. [JaBu96, GoHo00]). Dabei spielen auch kontextbezogene Wissensprozesse und deren Unterstützung innerhalb des Workflowsystems eine Rolle (vgl. [Abe⁺00, Sch⁺01c, Maus01, WeMa01]).

Abhängig von der Art der zu unterstützenden Prozesse können dabei Kontextinformationen aus vielen betrieblichen Teilbereichen und Kontextdimensionen benötigt werden, die entsprechend zu integrieren und darzustellen sind, um bei der Modellierung und Ausführung in angemessener Form bereitzustehen. Beispiele für kontextsensitiv zu erfassende Prozessparameter werden in Kap. 6.2 („Funktionale Validierung“, S. 175) gegeben. Andererseits können aus den in Workflowsystemen gespeicherten Prozess- und Aktivitätsdaten wesentliche Informationen über den aktuellen Arbeits- und Aufgabenkontext des Nutzers abgeleitet und für andere kontextbezogene Anwendungen bereitgestellt werden, etwa der Aufgaben- und Verantwortungsbereich sowie Zugriffsrechte des Nutzers (vgl. [GoHo00, Wenz98]).

3.4.11 Teamwork-Support, Workgroup Computing

Im Gegensatz zu *Prozessen*, welche stark repetitive, d. h. häufig wiederholte Tätigkeiten darstellen, wird unter *Teamwork* eine spontane, unregelmäßige Zusammenarbeit von Menschen verstanden. Der unstrukturierte, kreative Charakter von Teamarbeit setzt einer maschinellen Unterstützung prinzipielle Grenzen. Existierende IuK-Lösungen zur Unterstützung von Teamarbeit, die unter der Bezeichnung *Groupware* zusammengefasst werden, ermöglichen die Administration von Arbeitsgruppen, das computerunterstützte Austauschen von Nachrichten bzw. gemeinsame Bearbeiten von Materialien durch die Teammitglieder sowie die Vorbereitung, Moderation und Nachbereitung von Sitzungen, Meetings und Abstimmungen (vgl. [Mer⁺01]).

Der Kontextbezug beim *Groupware Computing* bezieht sich vor allem auf die Automatisierung der Konfiguration von Groupware-Werkzeugen durch automatische Erkennung von zu unterstützenden Vorgängen und den dabei relevanten Informationen. Beispielsweise muß einer klassischen Groupware-Anwendung explizit mitgeteilt werden, welche Personen Mitglieder des Teams sind. Diverse Forschungsansätze beschäftigen sich mit der Automatisierung dieses Vorganges durch Verfahren der automatischen Gruppenbildung bzw. der Erkennung von sozialen Situationen durch kontextsensitive Rechnersysteme (vgl. [Lei⁺00, SiSt01a]). So sollen z. B. Meetings oder Vorträge anhand verschiedener Kontextmerkmale automatisch erkannt werden, etwa durch die Kombination der Fakten, dass sich mehrere Personen über eine längere Zeit in einem geschlossenen Zimmer befinden und miteinander reden (Meeting) und dabei ein Beamer benutzt wird, wobei vor allem eine Person redet (Vortrag).

Basierend auf diesen Erkenntnissen können dann weitergehende Workgroup-Unterstützungsdienste konfiguriert werden, um etwa allen Meetingteilnehmern den Zugriff auf die gemeinsamen Dokumente des Teams, das aufgezeichnete Video des Vortrags sowie die dargestellten Folien zu ermöglichen, möglicherweise ebenfalls kontextindiziert wie in Kap. 3.4.1.3 („Information Retrieval“, S. 59) dargestellt (vgl. [Lamm91, LaNe92, Lam⁺94, Wer⁺01, Hen⁺02, Mei⁺04, Dey⁺99b]).

Kommunikationsaspekt	beeinflussender Kontextparameter
<i>Ansprechbarkeit</i> der Person	umgebende Personen, aktuelle Tätigkeit der Zielperson
angemessene <i>Lautstärke</i>	Umgebungsgeräusche, Hörvermögen der Zielperson
<i>Relevanz</i> der zu kommunizierenden Informationen	gegenwärtige Aufgaben und Probleme des Empfängers
<i>Interpretation</i> der kommunizierten Inhalte	Vorausgegangene Kommunikation, Kenntnisstand des Empfängers

Tabelle 10: Kommunikationskontexte

Neben der Erkennung unmittelbarer Realweltobjekte werden dabei auch weitergehende Informationen zu den einzelnen Realweltobjekten benötigt, die aus den verschiedensten Anwendungssystemen zusammengetragen und integriert werden müssen. Für Beispielszenarien muß an dieser Stelle wiederum auf Kap. 6.2 („Funktionale Validierung“, S. 175) verwiesen werden.

3.4.12 Mobile Informations- und Kommunikationsunterstützung

Wie schon aus deren Namen ersichtlich ist, stellt die Unterstützung der *Kommunikation* zwischen menschlichen Aufgabenträgern einen der hauptsächlichen Aufgabenbereiche von betrieblichen Informations- und Kommunikationssystemen dar. Nach HEINRICH stellt Kommunikation als Teilbereich des Informationsmanagements im betrieblichen Umfeld einen *erfolgskritischen Unternehmensaspekt* dar (vgl. [Hein99, S. 8]). Während unter Kommunikation generell der Austausch von Informationen zwischen Menschen und/oder Maschinen als Grundlage eines koordinierten Miteinander, d. h. als notwendige Voraussetzung für die Bildung sozialer Systeme verstanden wird (vgl. [StHa99]), wird an dieser Stelle besonders auf die durch maschinelle Systeme unterstützte Mensch-Mensch-Kommunikation eingegangen.

Ein wesentlicher Faktor für ein angemessenes Kommunikationsverhalten unter Menschen ist die Berücksichtigung der jeweiligen Situation vor und während des Kommunikationsvorganges. In einer *Face-to-Face*-Situation zwischen zwei Menschen sind diese Informationen implizit sichtbar und ermöglichen es einem Menschen, sich angemessen in der aktuellen Situation zu verhalten. Zu den dabei erwarteten Verhaltensweisen gehört etwa, nicht einfach drauflos zu reden, sondern zu warten, bis andere ihre Ausführungen beendet haben. Tabelle 10 gibt eine Auswahl der in diesem Zusammenhang relevanten Umweltparameter.

3.4.12.1 Kontextsensitive Telekommunikation

Durch betriebliche IuK-Systeme wird vor allem *Telekommunikation*, d. h. Kommunikation zwischen räumlich entfernten Partnern mittels dedizierter Nachrichtentransportsysteme unterstützt, von denen es eine enorme Vielfalt zur Unterstützung der unterschiedlichen Kommunikationsformen und -arten gibt (einen Überblick gibt [Häc⁺00]). In diesem Fall sind die in Tab. 10 für eine situationsbezogen adaptive Verhaltensweise benötigten Kontextparameter der Zielperson nicht bekannt und müssen durch explizite

Kontextinformationen dargestellt werden, um sowohl menschlichen als auch maschinellen Kommunikationspartnern das erwartete situationsbezogene Kommunikationsverhalten zu ermöglichen. Dies ist besonders bei der durch *Mobile Computing* ermöglichten allgegenwärtigen Kommunikation notwendig, die bei der durch häufige unpassende Anrufe schnell eine Abwehrreaktion der Nutzer gegen die Benutzung von Mobiltelefonen an sich entstehen kann.

Beispiel 10 *Ein kontextsensitives Telefonverzeichnis listet nicht nur die Nummern der einzelnen Personen zugeordneten Telefonanschlüsse in statischer Form auf, sondern zeigt situationsbezogen die jeweils geeignetste Form, diese Person zu erreichen. Dazu gehören auch Informationen, ob die betreffende Person für den jeweiligen Anrufer überhaupt zu erreichen ist, was nicht der Fall ist, wenn sie z. B. in einem Meeting sitzt oder das Telefon ausgeschaltet hat. (vgl. [Lie⁺99, Sch⁺00, Sch⁺01b]).*

Neben der in Beispiel 10 dargestellten kontextbezogenen Vorbereitung des Kommunikationsvorganges kann auch dessen Durchführung kontextbezogen adaptiert werden. Dazu zählt die automatische Auswahl der geeignetsten Modalität. Mögliche Modalitäten stellen textuelle Nachrichten (Instant Messaging), akustische Kommunikation (Telefon, Voice Chat) oder eine Videokonferenz dar. Die Modalität der Kommunikation sollte mittels adaptiver Endgeräte frei wählbar und jederzeit an die aktuellen Verhältnisse angepasst werden können, wie dies etwa im EU-Projekt *BIB3R* beschrieben wird (vgl. [O.V.05a]).

Asynchrone Nachrichten können abhängig von der aktuellen Situation zugestellt werden, wie im *Message Logistics*-Ansatz des Fraunhofer Instituts für Software- und Systemtechnik realisiert. Dabei werden basierend auf der Auswertung von Situationsmodellen nur die in der aktuellen Situation relevanten Nachrichten an den Nutzer zugestellt und die irrelevanten Nachrichten für eine spätere Zustellung aufgehoben (vgl. [Mei⁺04]). Für solche Entscheidungen werden umfassende Informationen über die aktuelle Situation benötigt, u. a. über die anwesenden Personen, geplante Vorgänge sowie die von diversen Groupware- und Workflowsystemen gelieferten Kontextinformationen.

3.4.12.2 Kontextsensitive Face-to-Face-Kommunikation

Auch die direkte Kommunikation von Angesicht zu Angesicht kann kontextbezogen unterstützt werden. Problematisch hierbei ist insbesondere die Tatsache, dass die eventuell benötigten Unterlagen bzw. Notizen insbesondere bei spontanen Treffen nur in den seltensten Fällen verfügbar sind und daher in vielen Fällen wieder auf klassische Kommunikationsmedien wie Email ausgewichen werden muß. Ferner fehlt eine Möglichkeit zur Aufzeichnung der ausgetauschten Inhalte. Kontextsensitive Gesprächsassistenten machen zu besprechende Inhalte im Rahmen eines kontextsensitiven Alarms verfügbar, sobald sich die betreffende Person, mit der diese Inhalte zu besprechen sind, in der Nähe bzw. am Telefon befindet. Dazu können neben PDAs *Augmented Reality*-Technologien verwendet werden. Die genannten Geräte können die mündlich ausgetauschten Informationen auch aufzeichnen und für eine spätere Auswertung kontextbezogen indizieren (vgl. Kap. 3.4.11 („Teamwork-Support, Workgroup Computing“, S. 75)).

Zur Unterstützung sämtlicher genannter Kommunikationsvorgänge werden nicht nur Ortsinformationen, sondern auch Informationen über die betreffenden Personen, zu klärende Fragestellungen sowie daran beteiligte Dokumente bzw. zumindest eine Verknüpfung auf den Dokumentenbestand des Nutzers benötigt, um dort nach den gewünschten Dokumenten zu suchen.

3.5 Technologische Effekte einer automatisierten Versorgung von betrieblichen Aufgabenträgern mit Kontextinformationen

Nachdem im vorigen Kapitel Nutzungsmöglichkeiten für umfassende, aus diversen Datenquellen entnommene Kontextinformationen im betrieblichen Umfeld vorgestellt wurden, werden in diesem Unterkapitel die dabei entstehenden wirtschaftsinformatischen Effekte beleuchtet.

3.5.1 Erhöhung der Informationsbasis über Realweltobjekte

Der wohl naheliegendste Effekt in diesem Zusammenhang ist der bessere Informationsgrad über Personen, Objekte und Vorgänge im Kontext. Die genaue Kenntnis von deren realweltbezogener Situation ermöglicht eine angepasstere Berücksichtigung dieser Realweltobjekte in betrieblichen Leistungsprozessen und damit eine bessere Dienstleistung. Als Beispiel seien die vielfältigen ortsbasierten Dienste (engl. *Location-Based Services*), aber auch die in Kap. 6.2 („Funktionale Validierung“, S. 175) dargestellten Realweltobjektinformationen nutzenden Anwendungen genannt.

3.5.2 Teilautomatisierung der Erkennung und Handhabung von Problemen

DRESBACH versteht unter einem Problem einen „*Zustand der Desorientierung*“, d. h. einen Zustand der ungenügenden Information bzw. Kontrolle über einen Realweltaspekt (vgl. [Dres99, S. 76]). Den ersten Schritt zur Lösung eines Problems stellt daher die Analyse des Problems in seinem originalen Umfeld, d. h. seinem Kontext dar. Dazu müssen Informationen über das Problem und dessen Umstände recherchiert werden, was mitunter recht zeitaufwändig ist.

Während die klassischen betriebswirtschaftlichen und wirtschaftsinformatischen Verfahren vor allem zur möglichst effizienten Steuerung bzw. Optimierung *geplanter* Abläufe mittels geeignet abstrahierter Sichtweisen konzipiert sind, werden zur Handhabung von Störungen, welche *ungeplante*, d. h. außerhalb des Planungshorizontes stattfindende Ereignisse darstellen, immer noch menschliche Aufgabenträger benötigt, da nur diese dank ihrer realweltbezogenen Sichtweisen die oft externen, realweltbezogenen Ursachen von Störungen erkennen und dank ihrer flexiblen, intelligenten Problemlösungsfähigkeit beheben können. Daher wird die Handhabung von Problemen auch in nächster Zeit nicht vollautomatisiert werden können (vgl. dazu auch Kap. 3.5.4 („Erhöhung des Automatisierungsgrades“, S. 82)), jedoch können durch die in dieser Arbeit entwickelten *Context Maps* zumindest einige weitere realweltbezogene Einflußfaktoren auf das Verhalten von Prozessen modelliert und damit semantische und teilweise sogar pragmatische Elemente des betreffenden Analysemodells abgebildet werden. Damit stellen *Context Maps* einen Schritt auf dem Weg zur Realisierung der Vision der Wirtschaftsinformatik, das betriebliche Geschehen zumindest sinnhaft vollzuautomatisieren, dar (vgl. [Mert03, S. 49]).

3.5.3 Effizientere Gestaltung der Mensch-Maschine-Schnittstelle

Maschinelle Aufgabenträger kommunizieren mit menschlichen und anderen maschinellen Aufgabenträgern sowie mit der Umwelt der Umgebung, wozu jeweils angepasste Schnittstellen verwendet werden (vgl. Abb. 26 auf der gegenüberliegenden Seite). Die Mensch-Maschine-Schnittstelle (engl. *Human-*

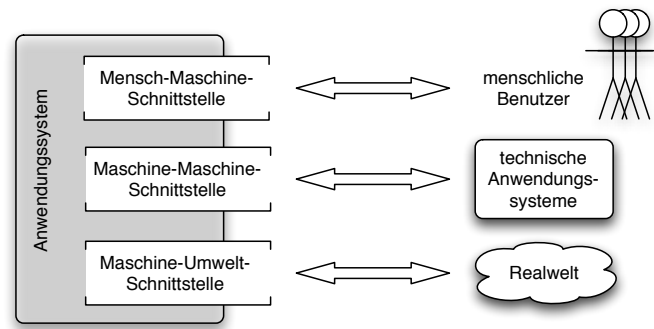


Abbildung 26: Arten von Eingabekanälen in Computeranwendungen

Computer-Interface, HCI) stellt dabei die Schnittstelle zur Kommunikation, d. h. dem Informationsaustausch zwischen menschlichen und maschinellen Aufgabenträgern dar.

Klassische Computersysteme verfügen lediglich über Eingabekanäle zu menschlichen Benutzern und anderen Anwendungssystemen und über keine Schnittstelle zur Umwelt, wodurch die benötigten Realweltinformationen manuell von menschlichen Nutzern eingegeben werden müssen, was die Mensch-Maschine-Schnittstelle stärker als notwendig belastet (vgl. Abb. 27(a) auf der nächsten Seite). In diesem Zusammenhang wird auf das in Kap. 1.1.1 („Vision: Assistenz des Menschen in der Informationsgesellschaft“, S. 3) dargestellte Problem der begrenzten Aufmerksamkeit, welche menschliche Nutzer bereit sind, einem Computersystem zu widmen, verwiesen. Beim kontextsensitiven Paradigma wird daher ein weiterer Eingabekanal für den *direkten* Austausch des Anwendungssystems mit der Umwelt hinzugefügt, wodurch die Mensch-Maschine-Schnittstelle auf die wirklich von Menschen einzugebenden Informationen verschlankt und dank der verfügbaren Kontextinformationen auf vielfältige Art adaptiert werden kann (vgl. Abb. 27(b) auf der nächsten Seite, [Hong01]).

3.5.3.1 Verschlinkung

Die Verschlinkung der Mensch-Maschine-Schnittstelle macht sich u. a. in einer Vereinfachung der von Unternehmensanwendungen verwendeten Eingabeformulare, die oft durch eine enorme Komplexität gekennzeichnet sind, bemerkbar. Einige der dort erwarteten Informationen können durchaus anderen Anwendungssystemen bekannt sein, etwa die Identität oder der augenblickliche Standort des Benutzers, der bearbeitete Prozess, der laufende Arbeitsschritt oder Informationen über den Kunden. Das wiederholte Ausfüllen solcher Informationen in verschiedenen Anwendungen führt schnell zu Frust bei den Anwendern, da die Anwendung als nicht „mitdenkend“ erlebt wird und den Benutzer dadurch zu unnötigen Eingaben zwingt. Das simple Anbieten bereits getätigter Eingaben stellt nicht nur ein Sicherheitsrisiko dar, sondern führt oft auch nicht zu den erwünschten Vereinfachungen der Nutzerschnittstelle, da deren Komplexität dadurch eher noch zunimmt. Ein kontextbezogenes Vorausfüllen gewisser Formularfelder hat jedoch das Potenzial, solche Eingaben ohne zusätzliche Komplexität sichtbar zu reduzieren (vgl. [Gos⁺03]). Die mittels *Context Maps* ermöglichte systematische Beschaffung sämtlicher benötigten Kontextinformationen kann sogar dazu führen, dass die Eingabemaske u. U. überhaupt

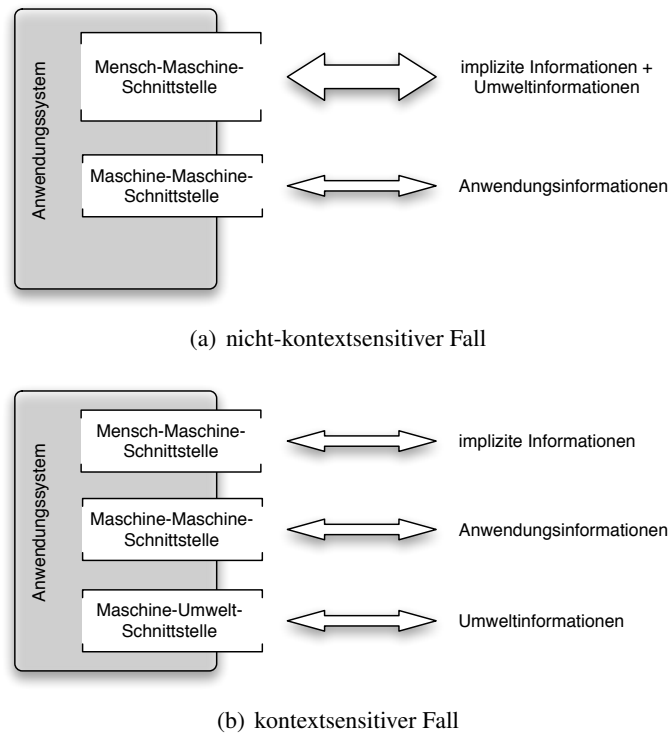


Abbildung 27: Eingabekanäle in Computeranwendungen im kontextsensitiven und nicht-kontextsensitiven Fall

nicht mehr angezeigt werden muß, wodurch ein automatisches Anwendungsverhalten erreicht wird (vgl. [Abo⁺97, Dav⁺01]).

3.5.3.2 Adaption

Die Adaption der Mensch-Maschine-Schnittstelle ist einer der am besten erforschten Bereiche kontextbezogener Adaptionen. Dabei können eine ganze Reihe von Adaptionen hinsichtlich diverser Kontextdimensionen erfolgen, wobei die dabei erzielten Effizienz- und Ergonomieverbesserungen durchaus signifikant sein können (vgl. [Brus97]). Die wichtigsten Adaptionsansätze werden im folgenden vorgestellt.

Adaption hinsichtlich physikalischer Kontextparameter. Kontextbezogene Adaptionen im Bereich der Nutzerschnittstelle beziehen sich zumeist auf den physikalischen Kontext. Dazu gehören Adaptionen der Lautstärke von Tonsignalen abhängig von der Lautstärke der Umgebungsgeräusche und in der Nähe befindlicher Personen. So ist es für ein Gerät z. B. sehr unangemessen, in einem Zimmer, in dem sich andere Menschen befinden oder gar schlafen, bei Empfang einer Textnachricht lautstark zu klingeln. Auch die Schriftgröße kann beim Lesen in unruhiger Umgebung vergrößert dargestellt werden, wodurch die Lesbarkeit der Texte verbessert wird (vgl. [Sch⁺99a]).

Adaption hinsichtlich nutzerbezogener Kontextparameter Die Adaption hinsichtlich der Identität und den Präferenzen des aktuellen Nutzers wird als *Personalisierung* bezeichnet. Die Berücksichtigung des individuellen Kenntnisstandes, Kompetenzprofils, der Aufgabenbereiche, Zugriffsrechte und Präferen-

zen des Nutzers ermöglicht die individuelle Gestaltung von Benutzungsschnittstellen und des Anwendungsverhaltens und damit eine effektivere Interaktion des Nutzers mit „seinen“ Werkzeugen. Ein Beispiel stellt die vergrößerte Darstellung von Schriften für alte oder sehbehinderte Menschen dar.

Die Berücksichtigung der *Rolle* des Nutzers im betrieblichen Umfeld ermöglicht das Ein- bzw. Ausblenden von Anwendungsfunktionen gemäß den Zugriffsrechten des Nutzers (vgl. [O.V.99]). Die gezielte Modifikation von Anwendungsoberflächen hinsichtlich des Erfahrungsstandes der Benutzer ermöglicht ein flüssigeres Arbeiten sowohl für Anfänger als auch für fortgeschrittene Nutzer, indem für Anfänger einführende Informationen und für Experten Abkürzungen zu häufig verwendeten Funktionalitäten dargestellt werden (vgl. [Gosl01, S. 26]).

Adaption hinsichtlich technischer Kontextparameter Hinsichtlich technischer Parameter ist ebenfalls eine breite Palette von Adaptionen der Nutzerschnittstelle von Computeranwendungen denkbar. Heutzutage werden Computersysteme von Menschen vor allem per Tastatur, Maus und Bildschirm bedient, was eine im tatsächlichen Nutzeralltag eingeschränkte *Dialogfähigkeit* bedeutet, da zur Interaktion mit Anwendungssystemen i. d. R. die gegenwärtige Tätigkeit in der Realwelt unterbrochen werden muß, um sich explizit der Interaktion mit dem Computer zu widmen. Die in Kap. 3.3 („Kontextbezogene betriebliche Anwendungssysteme“, S. 53) vorgestellten Konzepte erlauben durch Nutzung tragbarer Nutzerschnittstellen und Endgeräte die Erweiterung dieses Interaktionsschemas hin zu einer *situativen Interaktion* mit Computersystemen während realweltbezogener Tätigkeiten (vgl. [LeKr02]). Gerade diese mobilen Endgeräte sind jedoch durch starke Einschränkungen hinsichtlich der Ergonomie ihrer Nutzerschnittstelle gekennzeichnet, wodurch sie aufgrund der dadurch eintretenden Ermüdungserscheinungen bei längerem Gebrauch nicht wirklich dauerhaft eingesetzt werden können. Dies betrifft vor allem zu kleine Bildschirme, fehlende Tastaturen oder etwa eine unergonomische Haltung beim Benutzen dieser Geräte.

Beim *Nomadic Computing* werden daher eine Reihe von in der Nähe befindlichen stationären Geräten zur ergonomischeren Interaktion spontan für einen mobilen, d. h. beweglichen, aber zur Zeit nicht in Bewegung befindlichen Nutzer verfügbar gemacht. Dazu gehören hochauflösende stationäre Bildschirme, Tastaturen, Tablet-PCs, in der Nähe befindliche netzwerkfähige Drucker bzw. Geräte zur Audioein- und -ausgabe. Ziel einer gerätebezogenen Adaption der Nutzerschnittstelle ist die Erschaffung von sogenannten *composite devices*, also aus mehreren Geräten zusammengesetzter, aber trotzdem kohärenter Nutzerschnittstellen, welche die Informationen vom und zum Nutzer bestmöglich transportieren (vgl. [Wan⁺96, Bru⁺00, Pha⁺01, Pon⁺01, Rom⁺02, Hes⁺02, Jons02]). Dazu werden umfassende Informationen nicht nur über die technische Umgebung, sondern auch über die darzustellenden Inhalte und die daraus entstehenden Interaktionsmodalitäten benötigt.

3.5.4 Erhöhung des Automatisierungsgrades

Automatisierung bedeutet, formalisierbare menschliche Operationen durch selbsttätig agierende maschinelle Aufgabenträger ausführen zu lassen (vgl. [FeSi84]). Die maschinelle Erledigung von Tätigkeiten erfolgt dabei zumeist mit einer höheren Geschwindigkeit, Qualität, Präzision, u. U. auch Kostenstruk-



Abbildung 28: Klassifikation der durch Automatisierung zu unterstützenden Aufgabenbereiche hinsichtlich ihres Innovationsgrades

tur¹⁴ und Dokumentation¹⁵, aber einem stark eingeschränkten Innovationspotenzial, d. h. automatisierte Prozesse können sehr einfach wiederholt, aber nur relativ aufwändig verändert werden (vgl. [Fuc⁺99, S. 240]).

Da Arbeit nicht nur eine lästige Pflicht, sondern auch eine notwendige soziale Plattform zur Integration des Menschen in die Gesellschaft darstellt, kann die *vollständige* Ersetzung von menschlichen durch maschinelle Aufgabenträger kein ernsthaftes Entwicklungsziel der Wissenschaft darstellen. Dieses moralische Argument wird untermauert durch die Ergebnisse der nur bedingt erfolgreichen Versuche der Vollautomatisierung betrieblicher Produktionsabläufe in den späten 1980er Jahren (auch bekannt als Computer-Integrated-Manufacturing), welche deren technische Grenzen zur gegebenen Zeit offenbarten. Diese Versuche führten zu der Erkenntnis, dass maschinelle Aufgabenträger nicht in der Lage sind, die in der Realität unglaublich vielfältigen Ausnahmesituationen, die einen kreativen Problemlösungsprozess erfordern, angemessen handzuhaben (vgl. [Sied99], [Fuch97], [Kön⁺95], [KrSi99, S. 368 ff]). Dies ist auch für die nahe Zukunft nicht absehbar. Menschliche Aufgabenträger sollen daher nicht *ersetzt*, sondern lediglich von repetitiven, langweiligen oder gefährlichen Tätigkeiten aus dem linken Bereich des in Abb. 28 dargestellten Aufgabenspektrums *befreit* werden, damit sie ihre kostbare menschliche Intelligenz und Kreativität, die bisher nicht durch Maschinen ersetzt werden kann und soll, für innovative, kreative, interessante, anspruchsvolle oder entspannende Aufgaben aus dem rechten Bereich von Abb. 28 widmen können, die von maschinellen Aufgabenträgern weitestgehend zu *unterstützen* sind (vgl. [Wol⁺99, S. 314]).

Dabei ist eine Überautomatisierung, welche die kreativen Prozesse von Natur und Mensch unterdrücken kann, auch im linken Aufgabenbereich zu vermeiden, d. h. die Unternehmensorganisation muß so gestaltet werden, dass die Nutzer sowohl in ihrer Kreativität gefördert, d. h. in ihren Tätigkeiten weitestgehend durch maschinelle Aufgabenträger unterstützt werden, dabei jedoch einen Überblick über die ablaufenden Prozesse behalten und diese flexibel und veränderbar gestalten können. Daher geht es nicht einfach nur um eine Ersetzung von bisher durch den Menschen ausgeführten Tätigkeiten, sondern um „das viel komplexere Problem der Konzipierung und gesellschaftlichen Realisierung neuer Tätig-

¹⁴Personalkosten stellen mit 10-30 Prozent der betrieblichen Ausgaben einen erheblichen Kostenfaktor in europäischen Unternehmen dar und können durch Automatisierung reduziert werden (vgl. [Kutz03, S. 20]).

¹⁵Die Dokumentation eines von einem menschlichen Aufgabenträger erledigten Vorganges stellt einen expliziten, oft aus Zeitgründen vernachlässigten Vorgang dar. Da bei maschinell ausgeführten Tätigkeiten sämtliche benötigten Informationen bereits expliziert sind, können diese Vorgänge automatisch und sehr detailliert dokumentiert werden (vgl. Kap. 3.4.4 („Dokumentation/Logging“, S. 67)).



Abbildung 29: Zusammenhang von Pro- und Reaktivität

keitsprofile und Organisationsstrukturen infolge der Automatisierungsmaßnahmen“ (vgl. u. a. [Fuc⁺76, S. 25]).

Dazu werden maschinelle Assistenten, welche die Tätigkeiten des menschlichen Nutzers vorbereiten, unterstützen, und für spätere Auswertungen protokollieren, benötigt. Ziel der wirtschaftsinformatischen Tätigkeit stellt also die *sinnhafte* Vollautomation des betrieblichen Geschehens im Sinne einer möglichst umfassenden und *vollwertigen*, aber nicht *vollständigen* Assistenz der menschlichen Nutzer durch maschinelle Aufgabenträger dar.

Bezüglich des Themas dieser Arbeit lässt sich feststellen, dass Automatisierung die vollständige Kenntnis aller benötigten Informationen durch den maschinellen Aufgabenträger voraussetzt. Dies beinhaltet Information über die involvierten Realweltobjekte, deren Eigenschaften sowie Situation. Es ist jedoch kaum zumutbar, jede einzelne kontextsensitive Anwendung individuell zu konfigurieren und mit Kontextdaten auszustatten, besonders wenn diese in den Alltag normaler Endanwender integriert werden. Benötigt wird daher eine allgemein verwendbare Darstellung aller in diesem Zusammenhang benötigten Kontext- und Konfigurationsinformationen, die durch alle Assistenten und Anwendungen des Nutzers verwendet werden. Dadurch wird ein einheitliches, vorhersehbares Verhalten der elektronischen Assistenzsysteme erreicht, was aus Nutzersicht unbedingt wünschenswert ist. Das verwendete Kontextmodell muss daher so allgemein aussagefähig sein, dass es zum einen sämtliche benötigten Kontextinformationen enthält und zum anderen von allen Anwendungen verstanden und verwendet werden kann. Dies ist nur mit dem in dieser Arbeit entwickelten umfassenden und generischen Integrations- und Darstellungsmodell für Kontextinformationen möglich, weswegen dieses Modell zur Verbesserung diverser Automatisierungsformen und -aspekte beiträgt. Im folgenden werden der Nutzeraktion vor- und nachgelagerte Automatisierungsformen skizziert.

3.5.4.1 Reaktivität

Reaktive Systeme führen selbsttätig zu den stattfindenden Realweltvorgängen passende Aktionen aus, etwa als Reaktion auf Aktionen des Nutzers und zur Unterstützung dieser. Diese Schritte können sich sowohl in der elektronischen als auch in der Realwelt abspielen. Sehr viele der in Kap. 3.4 („Anwendungsfelder von Kontextinformationen in betr. IuK-Systemen“, S. 57) dargestellten Anwendungsfelder beinhalten reaktive Assistenzfunktionen für den Nutzer.

Reaktivität basiert prinzipbedingt auf der Kenntnis der durch den Nutzer durchgeführten Aktionen und der zu bearbeitenden Problemstellungen, die durch diverse Ereignisse in sämtlichen denkbaren Anwendungssystemen repräsentiert werden und in geeigneter Form in ein umfassendes Bild über die ablaufenden Vorgänge integriert werden müssen.

3.5.4.2 Proaktivität

Ein beachtlicher Teil der zur Problemlösung, d. h. zur Bearbeitung der *primären* Aufgabenstellung aufgewendeten Arbeitszeit wird für die Vorbereitung der eigentlichen Problemlösungsschritte, d. h. für *sekundäre* Aufgabenstellungen aufgewendet, z. B. zur Recherche von zur Problemlösung benötigten Informationen. Dieser Anteil ist sehr stark von der jeweiligen Aufgabenstellung, dem Kontext sowie dem impliziten Wissen des Bearbeiters abhängig und kann den Nutzer in nennenswertem Umfang beschäftigen.

Im Vergleich zur *reaktiven* Verhaltensweise, bei der ein Anwendungssystem lediglich auf auftretende Ereignisse reagiert und der Nutzer sowohl primäre als auch sekundäre Lösungsschritte selbsttätig durchführen muß, wird unter *Proaktivität* das eigenständige, vorausschauende Beschaffen und Anbieten von möglicherweise benötigten Informationen und Diensten und damit die Übernahme von sekundären Lösungsschritten durch einen elektronischen Assistenten verstanden, um zu erreichen, dass der Nutzer sich voll auf die primären Lösungsschritte konzentrieren kann (vgl. Abb. 29 auf der vorherigen Seite). EISENHAUER betrachtet die proaktive Informationsvorsorgung deshalb als wichtigsten Anwendungsbereich von Kontextinformationen (vgl. [EiKI01]).

Proaktivität ist in noch höherem Maße als Reaktivität von der Erkennung der aktuellen Situation, der dort ablaufenden Prozesse und Aktivitäten, der relevanten Problemstellungen sowie insbesondere den Schlußfolgerungen, welche Prozesse *als nächstes* stattfinden und welche Informationen dafür benötigt werden, abhängig. Die in dieser Arbeit entwickelten *Context Maps* stellen die für pro- und reaktive Assistenzsysteme benötigten multidimensionalen Kontextinformationen in einer bereits integrierten und daher einfach nutzbaren Form dar.

3.5.5 Unterstützung von Echtzeitverhalten

Unter Echtzeitverhalten wird die innerhalb sehr kurzer Zeit erfolgende und damit für einen menschlichen Nutzer verzögerungsfrei erscheinende Reaktion von Anwendungen auf Ereignisse in der Umwelt verstanden. Verzögerte Reaktionen auf Umweltereignisse stellen möglichst zu minimierende Ungenauigkeiten im betrieblichen Steuerungs- und Leistungsprozess dar, die in modernen Unternehmensmodellen wie dem von der GARTNER GROUP beschriebenen *Realtime Enterprise* adressiert werden (vgl. [Grou00]).

Beispiele für in Echtzeit reagierende betriebliche Anwendungen stellen die diversen zeitnah reagierenden Sicherheitsmechanismen dar, die nur bei Vorliegen aller notwendigen Sicherheitsmerkmale den Zugriff auf geschützte betriebliche Funktionen gewähren. Dazu gehören Produktionsanlagen, die automatisch anhalten, wenn sich Menschen zu nah bei ihnen befinden sowie Unternehmensanwendungen, die sensitive Unternehmensdaten nur dann anzeigen, wenn eine ausreichende Sicherheitslage sowohl in technischer als auch personeller Hinsicht gegeben ist, d. h. die verwendeten Endgeräte sämtliche benötigten Sicherheitsstandards beherrschen und sämtliche in der Nähe befindlichen Benutzer autorisiert sind, die Informationen zu einzusehen.

Grundlage echtzeitfähiger Anwendungssysteme stellen neben den angesprochenen echtzeitfähigen Anwendungen auch in Echtzeit bereitgestellte Kontextinformationen dar. Da die im betrieblichen Umfeld eingesetzten *Enterprise Application Frameworks* eher hinsichtlich Transaktionssicherheit und Ge-

samtdurchsatz und weniger auf echtzeitfähige Reaktionen optimiert sind, wird Echtzeitfähigkeit im betrieblichen Umfeld heutzutage eher bei Spezialanwendungen implementiert, da der dazu notwendige technologische Aufwand nur in besonderen Fällen ökonomisch zu rechtfertigen ist (vgl. u. a. [Gos⁺03, S. 3]). Ein *Realtime Enterprise* setzt also neben echtzeitfähigen Unternehmensanwendungen – auf die an dieser Stelle nicht weiter eingegangen werden soll – auf die automatisierte Erfassung von sämtlicher benötigter Umwelteinflüssen mittels kontextsensitiver Konzepte sowie die geeignete Darstellung dieser umfassenden Informationen mittels entsprechender Integrations- und Darstellungskonzepte, welche in dieser Arbeit entwickelt werden.

3.6 Wirtschaftliche Effekte einer automatisierten Versorgung betrieblicher Aufgabenträger mit Kontextinformationen

Die im vorigen Unterkapitel dargestellten technologischen und wirtschaftsinformatischen Effekte der Versorgung maschineller Aufgabenträger mit umfassenden, integrierten Kontextinformationen führen zu den im folgenden dargestellten wirtschaftlichen Effekten.

3.6.1 Engere Kundenorientierung

Die Kenntnis individueller Merkmale der Kunden des Unternehmens sowie von deren jeweiliger Situation ermöglicht die schnellere, bessere, umfassendere und angepasstere Erbringung von Diensten für diese Kunden und damit einen Wettbewerbsvorteil gegenüber einer nicht kontextbezogen arbeitenden Konkurrenz. Im Bereich der Kundenorientierung können diverse Nutzungsgebiete identifiziert werden, auf welche im folgenden separat eingegangen wird.

3.6.1.1 Mass Customization

Im Bereich *Mass Customization*, d. h. der individuellen Konfiguration einzelner, mittels eines hochflexiblen Serienherstellungsprozesses produzierter Produkte hinsichtlich einzelner Kundenwünsche ermöglichen kombinierte Kontextinformationen eine weitergehende Automatisierung und damit Kostensenkung (vgl. [II92]). Ein Kunde, der sich in einem entsprechenden Geschäft z. B. mittels einer Smartcard elektronisch identifiziert, kann dem Hersteller Zugriff auf Teile seines Nutzerprofils ermöglichen und damit die angepasste Dienstleistung automatisch konfigurieren.

Indem zu jedem Identifikationsverfahren wie Rabattkarten lediglich ein Verweis auf das durch *Context Maps* realisierte Nutzerprofil des Kunden hinterlegt werden muß, wird nicht nur die Identifikation des Kunden flexibilisiert, da sie nun mittels beliebig kombinierbarer Verfahren realisiert werden kann, gleichzeitig bleibt der Zugriff auf die personenbezogenen Informationen auch unter der Kontrolle des Verbrauchers, welcher die entsprechende *Context Map* verwaltet und insbesondere die Zugriffsrechte darauf festlegt. Gleichzeitig werden durch die strukturierte, semantisch aussagefähige Darstellung der nutzerbezogenen Informationen personalisierte Dienstleistungen wie *Mass Customization* weitgehend automatisiert und damit kosteneffektiviert.

3.6.1.2 C-Marketing

Werbung ist für demokratische, marktwirtschaftliche Grundprinzipien wie eine vollständige Marktübersicht aller Marktteilnehmer, die letztlich zum Anbieterwettbewerb führt, unverzichtbar. Der Nutzen einer Verbraucherinformation hängt dabei jedoch sehr stark von der gegebenen Situation des Dienstleistungsnachfragers ab. So ist ein Kunde erst dann an Informationen zu KFZ-Werkstätten in der Nähe interessiert, wenn sein Auto einen Schaden hat. Gleiches gilt für Werbung für Restaurants, die nur dann einen positiven und maximalen Effekt beim Verbraucher erzielt, wenn dieser auch tatsächlich auf der Suche nach einem Restaurant ist.

Die heutige Werbung ist durch ihre situationsunspezifische und mengenmäßige Verbreitung für die meisten Konsumenten hinsichtlich ihrer konkreten Beschaffungsvorhaben am Markt nicht mehr zweckdienlich, wird daher von vielen Personen abgelehnt und ist damit für den Werbetreibenden wirkungslos bzw. sogar schädlich, z. B. durch Haßgefühle und Boykott von zu aufdringlichen Firmen.

Das Internet ermöglichte es, den bei Printmedien auftretenden *Marketingkompromiss*, entweder eine hohe Zahl von Adressaten mit unspezifischen Informationen oder eine geringere Zahl von Adressaten mit spezifischen Informationen zu versorgen, aufzubrechen, benötigt dazu aber detaillierte Informationen über den individuellen Bedarf der einzelnen Konsumenten (vgl. [Mer⁺01]). Nach wie vor nicht zufriedenstellend gelöst ist die Frage, wo und wie die Kundeninformationen in dem für ein angemessenes Marketing benötigten Detaillierungsgrad beschafft werden können.

Erst die systematische und strukturierte Erfassung einer breiten Palette von Kontextinformationen ermöglicht eine situativ, qualitativ sowie quantitativ angepasste Form der Verbraucherinformation. Zum einen können durch die automatisierte Identifikation des Kunden (der in diesem Zusammenhang auch unter einem temporären Aliasnamen auftreten kann) Informationen nur zu den vom Nutzer tatsächlich gewünschten Produkten präsentiert werden, und dies auch nur in Situationen, in denen der Nutzer einen Bedarf für diese Informationen hat.

Entsprechend sind Nutzer- und Kontextinformationen essentielle Komponenten bei der Gestaltung situationsspezifisch passender, individuell gestalteter und damit einen maximalen Informations- und Imageeffekt verursachender Verbraucherinformationen auf Seiten der Dienstleistungsanbieter, können von Dienstleistungsnachfragern aber auch zur Filterung unpassender Dienstleistungsangebote in der Anbahnungsphase von Dienstleistungen verwendet werden.

Ein Beispiel stellt der in Beispiel 1 auf S. 7 dargestellte *RestaurantFinder* dar. Hierbei können weitere Kontext- und Nutzerinformationen, etwa dessen Präferenzen hinsichtlich vegetarischer, fett- oder kohlenhydratarmer Diäten oder religiöse Hintergründe, etwa das Verbot, Rindfleisch, Schweinefleisch, unkoscheres bzw. Hunde- oder Katzenfleisch zu essen, berücksichtigt werden, die ein Nutzer, der Wert auf diese Dinge legt, in seinem Profil ablegen kann, damit sie von den entsprechenden Anwendungen verwendet werden können. Fernerhin würden diese Produktinformationen bei einem kontextbezogenen Marketing dem Kunden nur dann zugestellt werden, wenn er auf der Suche nach einem passenden Restaurant ist, d. h. wenn z. B. eine Restaurantsuchanwendung läuft. Durch diese Maßnahmen kann Werbung ihren lästigen Charakter verlieren und einen informativen Charakter zurückgewinnen, was ein Gewinn für alle Beteiligten wäre.

3.6.1.3 C-Marktforschung

Neben der kontext- und situationsbezogenen Zustellung von Verbraucherinformationen (C-Marketing) können kontextbezogene Technologien auch zur Analyse des Verbraucherverhaltens (C-Marktforschung) verwendet werden. Dabei werden analog zur Verhaltensbeobachtung, Experimenten und Panels sowie ihren jeweiligen Online-Varianten im Internet Kontext-Beobachtungen, Kontext-Experimente und Kontext-Panels eingesetzt, um detaillierte Informationen über das Verbraucherverhalten und den Produkt- und Dienstleistungsbedarf der Verbraucher im Alltag zu sammeln und als Basis kontextbezogener Dienstleistungen zu verwenden (vgl. [Merz02]). Dies muß selbstverständlich im Einklang mit den datenschutzrechtlichen Rahmenbedingungen erfolgen, d. h. der Nutzer muß die Kontrolle über seine Privatsphäre behalten und darf nicht ohne sein Einverständnis und auch nicht in unbegrenztem Umfang beobachtet werden.

Ein Beispiel wäre die Beobachtung, wann (anonymisierte) Kunden wo einkaufen gehen, um basierend auf diesen Informationen die Positionierung von Verkaufsstellen sowie des darin bereitgestellten Produktangebotes zu bestimmen. Diese Angaben werden heute noch vom Verkaufsstellenpersonal mit einer entsprechenden Subjektivität und Fehlerrate geschätzt. Bezüglich des Themas dieser Arbeit kann festgestellt werden, dass hierfür die Kombination einer Reihe von durchaus unterschiedlichen kontextbezogenen Informationen etwa über den Ort bzw. Weg des Nutzers, die dazu verwendeten Transportmittel, die eingekauften Produkte sowie ggf. wann und wo sie verbraucht werden, notwendig sind.

3.6.1.4 Kontextunterstützter Point-of-Sale

Die METRO GROUP demonstriert mit dem *Future Store* moderne betriebliche Anwendungen von kontextbezogenen Technologien im Verkaufsgeschäft (vgl. [Grou03, Mier03]). Dabei werden sowohl im Backend als auch Frontend kontextbezogene Technologien eingesetzt, um neben der Logistik auch den Einkaufsvorgang komfortabler, effektiver und effizienter zu gestalten, ohne dadurch die Vorteile des klassischen Ladengeschäftes (Herumstöbern im Geschäft, direktes Betrachten und Berühren der Waren) aufzugeben.

Durch die Kombination solcher elektronischer Verkaufssysteme mit kontextbezogenen Assistenzfunktionen auf der Nutzerseite ergeben sich erhebliche Komfortsteigerungen für den Verbraucher und damit Wettbewerbsvorteile für die Geschäfte. Ein Geschäft, vor dem der Kunde von seinem PDA daran erinnert wird, noch die zuhause fehlende Zahnpasta zu kaufen, wird mehr Umsatz machen als ein konventionelles Geschäft, an dem der Kunde unwissend vorbeiläuft. Dies verdeutlicht die Motivation auch für Anbieter, ihr Angebot für kontextbezogene Verkaufstechnologien zu veröffentlichen (vgl. [Hul⁺97, S. 10]).

Durch die Integration einer ganzen Reihe von Kontextinformationen und -dimensionen kann der Verkaufsvorgang noch weiter automatisiert werden. Vorausgesetzt, der Kunde hat eine elektronisch verfügbare, z. B. in einem PDA gespeicherte Einkaufsliste, wäre durch die Kombination von Einkaufsliste und elektronischem Verkaufssystem ein kontextsensitiv aufgewerteter Einkauf möglich, bei dem der Kunde vom ortssensitiven Verkaufssystem ggf. durch den Laden geführt wird, um seinen Einkaufszettel abzarbeiten. Zur Zeit nicht mehr vorrätige Artikel können zur späteren Lieferung nach Hause bestellt werden, wobei dann die Adresse zusammen mit einem Verbot, dorthin Werbung zuzustellen, ebenfalls automatisch verfügbar gemacht wird (vgl. Abb. 30 auf der nächsten Seite).



Abbildung 30: Personalisierte Verwaltung von Einkaufslisten im *Future Store* der METRO GROUP

3.6.2 Flexibleres Controlling

Die zur Steuerung von betrieblichen Leistungsprozessen eingesetzten Informationsflüsse, d. h. Aufträge und Belege für erbrachte Leistungen, können durch relevante Umweltinformationen ergänzt und so eine Verbesserung der Sachziele des Controllings bzw. der Unternehmensziele erreicht werden.

Strukturierte, detaillierte Kontextinformationen stellen eine notwendige Grundlage insbesondere für das Controlling von stark in die Realwelt eingebundenen Unternehmensbereichen und Betriebsprozessen dar. Dazu gehört der bereits in Kap. 3.4.8 („Lagerhaltung“, S. 72) vorgestellte Bereich der *Lean Production*, wobei insbesondere *standortübergreifende* Distributionsvorgänge von Ausgangsstoffen, Halb- und Fertigproduktion sehr stark von externen Störgrößen beeinflusst werden können und entsprechend hinsichtlich möglicher Umweltfaktoren geplant und durchgeführt werden müssen. Dazu müssen, wie ebenfalls bereits dargestellt, diverse Kontextinformationen aus verschiedenen Kontextinformationen miteinander integriert werden, wobei das in dieser Arbeit entwickelte umfassende Kontextmodell zumindest eine strukturierte Grundlage für ein entsprechendes kontextsensitives Controllingsystem darstellen kann.

3.6.3 Effizientere Wartung

Die Ausfallkosten teurer Anlagen fallen aufgrund der damit einhergehenden Ausfallzeiten auch bei deren Wartung an. Als besonders problematisch erweist sich dabei die Wartung defekter Geräte in abgelegenen Regionen der Erde, wo Personal zwar vorhanden, jedoch oft nicht mit den betreffenden Spezialkenntnissen ausgestattet ist, um komplexere Maschinen zu reparieren. Ein extremes Beispiel stellt militärisches Gerät dar, das außerhalb der eigenen Basis ausfällt und von den betreffenden Soldaten repariert werden muß (vgl. [O.V.04h]).

Kontextbezogene Technologien wie *Wearable Computing* und *Augmented Reality* ermöglichen Wartungsarbeitern die effektivere Arbeit an solchen Geräten, indem sie ihnen die passenden Informationen zur richtigen Zeit in das Sichtfeld einblenden bzw. einzelne Arbeitsschritte direkt am zu wartenden Objekt markieren (vgl. Kap. 3.3.1 („*Wearable Computing*“, S. 54) und 3.3.2 („*Augmented Reality*“, S. 55)).

Trotz dieser fortgeschrittenen Darstellungstechniken soll die Darstellung von Informationen an den Nutzer möglichst minimiert werden, denn auch *Augmented Reality* lenkt den Nutzer von seinen eigentlichen Tätigkeiten ab bzw. verdeckt einen Teil seines Sichtfeldes. Ihr volles Potenzial können solche

kontextbezogenen Assistenzsysteme daher erst entfalten, wenn ihnen neben den aufgaben- auch personenbezogene Informationen z. B. über den Kenntnisstand des betreffenden Technikers, die Geräte, die er bereits gewartet und die Arbeitsschritte, die er dabei ausgeführt hat, bekannt sind. Nur dann sind sie in der Lage, ihn mit den wirklich benötigten Informationen in der passenden Detaillierung zu versorgen. So muß untrainierten Wartungsarbeitern die Benutzung von Spezialwerkzeug vor deren erster Verwendung erklärt werden, während Experten nur noch informiert werden müssen, was sie wo damit tun sollen (vgl. Kap. 3.4.5 („E-Learning, Computer aided Instruction“, S. 68)).

3.6.4 Effektivere Nutzung der betrieblichen Wissensbasis

Die betriebliche Wissensbasis, welche sowohl das persönliche Wissen der einzelnen Mitarbeiter als auch organisatorisches Wissen umfasst, stellt einen der wichtigsten Einflußfaktoren auf die Wettbewerbsposition eines Unternehmens dar (vgl. [PiFr88, Pro⁺98]).

Um die individuellen Kenntnisse jedes einzelnen Mitarbeiters optimal z. B. im Rahmen einer Teambildung oder als Ansprechpartner für Detailfragen nutzen zu können, ist es notwendig, diese zu kennen (vgl. Kap. 3.4.3.1 („Organisationale Perspektive“, S. 65), Beispiel 7 auf S. 59). Dazu werden neben Einschätzungen durch Vorgesetzte zunehmend auch kontextbezogene Verfahren eingesetzt, um die Kompetenzen von Angestellten anhand ihres Verhaltens zu identifizieren. HEEREN und SIEHN analysieren mittels eines *XpertFinder* genannten Verfahrens die elektronische Kommunikation (Emails, Newsgroups) der Nutzer auf Stichwörter aus einer Themenontologie¹⁶, um Kompetenzprofile der Angestellten hinsichtlich der jeweiligen Themenfelder zu erstellen. Hilfesuche zu einem Themenfeld können an das *XpertFinder*-System gestellt werden, welches die Anfrage dann an die entsprechend geeignetesten Kompetenzträger weiterleitet, die dann selbst entscheiden können, auf die Anfrage zu reagieren bzw. ihre Identität preiszugeben (vgl. [HeSi01, Heer01]). Besonders attraktiv erscheinen solche Verfahren bei einer *unternehmensübergreifenden* Expertensuche, wo häufig keine sozialen Kontakte und entsprechendes Hintergrundwissen über die Angestellten existiert.

Es ist offensichtlich, dass der *XpertFinder*-Ansatz durch weitere Kontextinformationen ergänzt werden kann, um noch detailliertere Kompetenzprofile zu erstellen. Dies betrifft u. a. Länder, in denen sich ein Mitarbeiter für längere Zeit aufgehalten hat oder ebenfalls zu Themengebieten zugeordnete Projekte, an denen er gearbeitet hat und über die er praxisrelevantes Detailwissen verfügt. In diesem Zusammenhang können *Context Maps* zur Integration dieser Informationen hinsichtlich der Qualifikation von Mitarbeitern verwendet werden.

3.6.5 Leistungsbezogenere Vergütung

Unter *leistungsbezogener Vergütung* als Instrument zur Erhöhung der extrinsischen Motivation von Angestellten wird die Entlohnung der Mitarbeiter abhängig von ihren tatsächlich erbrachten Arbeitsleistungen verstanden (vgl. [FeZa00, Kres01]). Unter *Motivation* werden in diesem Zusammenhang Beweggründe, welche die Richtung, Intensität und Ausdauer des Verhaltens von Angestellten bestimmen, verstanden (vgl. [Vroo64b]). Grundlage dieser Ansätze ist die Theorie, dass Angestellte mit dem Ziel, ihre

¹⁶HEEREN und SIEHN sprechen von *Themenfeldern*, die in einem *Themenfeldbaum* organisiert sind (vgl. [HeSi01, S. 84]).

eigene Zufriedenheit zu maximieren, rational diejenige Alternative aus verschiedenen alternativen Verhaltensweisen auswählen, welche ihnen in ihrer individuellen Situation als die zielführendste erscheint (vgl. [Vroo64a]). Aufgabe des Managements ist es daher, eine Situation im Unternehmen zu erzeugen, welche die Angestellten dazu bringt, Arbeit *für* das Unternehmen als auch für ihre persönlichen Ziele produktiv zu erachten (vgl. [McGr60]). Ein Weg dazu ist das bereits angesprochene Verfahren, die Entlohnung der Angestellten von deren individueller Arbeitsleistung abhängig zu machen, wodurch aufgrund wahrgenommener Fairness und Anerkennung von Leistung bei der Entlohnung die Motivation und damit die Leistungsbereitschaft der Mitarbeiter erhöht wird.

Zu den verwendeten Metriken zur Erfassung der Arbeitsleistung zählen einem Angestellten direkt zuordbare, abzählbare Meßgrößen wie die mittels Stechuhren erfasste Aufenthaltszeit auf dem Betriebsgelände bzw. die Erfassung der tatsächlichen Arbeitszeiten in mobilen Einsatzszenarien durch kontextbezogene Lokalisierung der Außendienstmitarbeiter, das Zählen von ausgeführten Arbeitsschritten bei der tayloristischen Fließbandarbeit mit entsprechendem Lohnabzug pro Fehler sowie die Erkennung von externen Störeinflüssen auf das erhaltene Arbeitsergebnis. Es werden jedoch auch auf eine ganze Arbeitsgruppe bezogene Metriken wie der von einer Abteilung erwirtschaftete monetäre Überschuß verwendet, um das Gruppendenken (Teamgeist) zu fördern. Ebenso verwendet werden können die von Workflowsystemen ermittelten Auftragsbearbeitungsstände (vgl. [FeSi98, S. 79 f]). Die dadurch von den Angestellten wahrgenommene Instrumentalität von Arbeitsleistungen aufgrund von fairen Belohnungsmechanismen führt zu einer starken extrinsischen Motivation (vgl. [Her⁺59, McGr60, Vroo64a, Hous71]).

Die Verwendung von aufgrund praktischer Restriktionen nur wenigen und daher für die Bewertung der tatsächlichen Arbeitsleistung nur bedingt repräsentativen Indikatoren führt jedoch dazu, dass die Angestellten ihr Verhalten starr an den Vorschriften bzw. auf die Optimierung der entsprechenden Indikatoren orientierten, was wiederum zu suboptimalen Arbeitsergebnissen führt, da zwar die Leistungsindikatoren erfüllt werden, andere wichtige, jedoch nicht durch Indikatoren erfasste Aspekte wie Teamfähigkeit nicht belohnt und damit von den Angestellten vernachlässigt werden.

Adressiert werden kann dieses Problem durch die Erfassung eines breiteren Spektrums von in der Realwelt erbrachten Arbeitsleistungsindikatoren, indem z. B. durch Kontextinformationen beschriebene realweltbezogene Aspekte der Arbeitsleistung einzelner Arbeitskräfte zur Beurteilung der Arbeitsleistung der einzelnen Mitarbeiter herangezogen werden. Diese würden objektiviertere und dadurch fairere, da tatsächlich leistungsbezogene extrinsische Motivationstrategien ermöglichen.

Dazu zählt, wie häufig ein Mitarbeiter mit Anfragen konfrontiert wird (dies spiegelt seinen Wert in der Organisation wider), die durchschnittliche Dauer bis zur Beantwortung von Anfragen oder EMailen bzw. wie lange einzelne Vorgänge bei den jeweiligen Mitarbeitern lagern, ehe sie bearbeitet werden. Auf diese Weise können nicht nur ein effektiveres Arbeiten der Mitarbeiter gefördert, sondern auch Schwachstellen in der betrieblichen Ablaufstruktur identifiziert und entsprechende Maßnahmen getroffen werden (vgl. Kap. 3.6.2 („Flexibleres Controlling“, S. 88)).

Zu beachten ist in diesem Zusammenhang, dass solche Methoden zumindest nach dem deutschen Rechtsverständnis nicht zulässig sind, da sie eine für die primäre Leistungserbringung nicht notwendige Überwachungen der Privatsphäre der Mitarbeiter am Arbeitsplatz darstellen (vgl. §3a, §4, §6a-c, insbesondere jedoch §6a, Abs. 1 und 2 Bundesdatenschutzgesetz). Fernerhin ist dieses Vorgehen auf jeden Fall

mit den Arbeitnehmern bzw. dem Betriebsrat abzustimmen. Ebenso ist die Beurteilung der Arbeitsleistung von Personen allein auf maschinell erhobenen Informationen nicht überall gesetzlich zugelassen.

3.6.6 Erhöhung der Transparenz

Im Gegensatz zum technischen Transparenzbegriff, unter welchem die selbsttätige und damit keine weitere Aufmerksamkeit auf sich ziehende Realisierung von Teilzielen verstanden wird, d. h. eine Unsichtbarkeit der diesbezüglichen internen und externen Vorgänge, wird unter Transparenz im wirtschaftlichen Sinne die möglichst vollständige Kenntnis der internen Struktur von Vorgängen verstanden, d. h. eine Unsichtbarkeit der Oberfläche, wodurch die internen Vorgänge sichtbar werden.

Umfassende Kontextinformationen spielen eine tragende Rolle bei der Transparentmachung von Prozessen, da sie deren realweltbezogene Aspekte und damit wichtige Merkmale des Prozessablaufes elektronisch beschreiben und damit auch räumlich und zeitlich entfernten Beobachtern zugänglich machen. Dies ermöglicht nicht nur das verbesserte Erkennen von Problemen, sondern auch deren rechtzeitige Prognose und Vorbeugung sowie die individualisierte Erbringung und Abrechnung von Dienstleistungen.

Beispiel 11 *Eine Autovermietung kann Kunden, die einwilligen, den Standort ihres Mietwagens sowie dessen Geschwindigkeit überwachen zu lassen, Rabatte bei der Autoversicherung erteilen, weil sie dadurch in der Lage ist, gemäßigte und besonnene Fahrer, die sich nicht in gefährlichen Gegenden aufhalten, zu erkennen sowie gestohlene Fahrzeuge sofort zu lokalisieren.*

3.6.7 Effektivere Problemidentifikation

Probleme in Prozessen entstehen nicht nur durch zu optimierende prozessinterne Schwachstellen, sondern auch durch störende externe Einflüsse auf diese Prozesse. Diese müssen durch eine Beobachtung des ablaufenden Prozesses sowie dessen Kontext erkannt und identifiziert werden, wozu neben den prozessinternen Informationen umfassende Kontextinformationen der betreffenden Prozesse benötigt werden.

Durch den Abgleich der Prozesskontextinformationen mit den Prozessinformationen kann der menschliche Aufwand zur Erkennung und Behebung von Problemursachen verringert werden, da die Prozesskontextinformationen oft bereits Hinweise auf die realweltlichen Ursachen der im Prozessablauf auftretenden Unregelmäßigkeiten enthalten.

So kann die Veränderung des Arbeitsgeräusches einer Maschine Rückschlüsse über mögliche Probleme mit dieser Maschine ermöglichen. Gleiches gilt für die Innentemperatur von Geräten bzw. Räumen, in welchen sich die betreffenden Geräte befinden, welche zu einem Ausfall bzw. Fehlverhalten der betreffenden Geräte führen kann.

3.6.8 Verringerung von Prozessdurchlaufzeiten

Die Verfügbarkeit von umfassenden Kontextinformationen führt im Zusammenspiel mit der bereits besprochenen Transparenz von Prozessen zu deren detaillierterer Analyse und Optimierung, was sich in verkürzten Prozessdurchlaufzeiten widerspiegelt.

Dies umfasst die Identifikation von Engstellen, d. h. sogenannten *Flaschenhälsen* und Umwegen im tatsächlich ablaufenden Prozess, welche in der Planung dieses Prozesses aufgrund eines fehlenden Realweltbezuges nicht sichtbar waren und daher nicht berücksichtigt werden konnten.

Als Beispiel sei die Identifikation von „*Prozessbremsen*“, d. h. Mitarbeitern, die Aufträge besonders langsam abarbeiten, genannt. Diese Informationen können prinzipbedingt nicht in der Planung von Abläufen berücksichtigt werden, da sie erst zur Laufzeit bekannt werden. Die Verfügbarkeit von umfassenden Kontextinformationen ermöglicht in diesem Zusammenhang eine tiefere Recherche über die Ursachen, etwa die Unterscheidung ob der betreffende Mitarbeiter mit anderen relevanten betrieblichen Dingen beschäftigt und damit überlastet oder aus anderen Gründen zu langsam ist. In beiden Fällen kann eine Entscheidung über Optimierungsmaßnahmen durch die umfassenden Kontextinformationen fundierter und gezielter getroffen werden, im Beispiel etwa hinsichtlich möglicher Weiterbildungsmaßnahmen für den Mitarbeiter, die auch gleich durch auf die gleiche Weise identifizierte effizientere Mitarbeiter durchgeführt werden könnte.

3.7 Kontextbezogene Dienstleistungsformen

Der auch als *tertiärer Sektor* einer Volkswirtschaft bezeichnete Dienstleistungssektor spielt im modernen Wirtschaftsgeschehen eine immer dominierendere Rolle, denn heute werden statt *Produkten* zunehmend *Problemlösungen* verkauft, deren zentrales wertschöpfendes Element nicht mehr die eigentlichen *Sachgüter*, sondern die von ihnen erbrachten *Dienste* darstellen, d. h. Sachgüter stellen letztlich nur ein Mittel zur Erreichung der eigentlich verkauften Problemlösungen oder *Dienstleistungen* des Anbieters und nicht mehr dessen zentrale Geschäftsinhalte dar. Das zentrale Element einer *Dienstleistung* stellt im Gegensatz zur *Sachleistung* entsprechend nicht die Übertragung des Eigentums an einem Produkt dar, sondern die Durchführung einer Leistung mittels Anwendung der Funktionalität eines Produktes am Kunden oder einem diesem gehörenden Objekt (vgl. [Bode99]).

Analog zu anderen Bereichen der Wirtschaftsinformatik lassen sich Dienstleistungen hinsichtlich diverser Kriterien klassifizieren. Für diese Arbeit relevant ist die Unterscheidung zwischen *standardisierten* und *individuellen* Dienstleistungsformen. Während bei ersten die Dienstleistung in der Annahme gleichartiger Kundenanforderungen für einen fiktiven Durchschnittskunden konzipiert und erstellt wird, orientieren sich individuelle Dienstleistungsformen an einem konkreten Individuum und werden speziell für dieses entworfen und realisiert.

Einen weiteren wichtigen Begriff stellt die *Service-Qualität* dar, welche die Güte der Dienstleistung, d. h. den Grad der Erfüllung der Erwartungen des Kunden an die Dienstleistung beschreibt. Ein wichtiger Bestandteil des von PARASURAMAN, ZEITHAML and BERRY entwickelten *RATER*-Systems zur Bewertung der Qualität einer Dienstleistung stellt neben der Zuverlässigkeit, Freundlichkeit, Kompetenz und Schnelligkeit auch das *Einfühlungsvermögen* (engl. *empathy*) des Dienstleisters dar, was im Rahmen einer weitergehenden Automatisierung von Dienstleistungen in der Zukunft auch für die maschinellen Aufgabenträger des Dienstleisters gelten muß, dann allerdings *Kontextsensitivität* genannt werden sollte (vgl. u. a. [Par⁺88]). Dies ist notwendig, da Dienstleistungen aufgrund ihrer typischen Einbettung in die betrieblichen und realweltbezogenen Abläufe beim Kunden in einem wesentlich stärkeren Maß von dessen individueller Situation abhängen als dies im ersten oder zweiten volkswirtschaftlichen Sektor

Allgemeine Bezeichnung	Dienstleister-Sichtweise	Nachfrager-Sichtweise
Anbahnung	Leistungsbereitstellung	Produkt-/Partnersuche
Vereinbarung	Leistungsvereinbarung	Auftragserteilung
Abwicklung	Leistungserbringung	Dienstleistungskonsum

Tabelle 11: Phasen von Dienstleistungen

üblich ist, wo es „nur“ um die Produktion bzw. den Verkauf von Rohstoffen und Produkten geht. Der damit einhergehende relativ hohe Freiheitsgrad bei der Gestaltung konkreter Dienstleistungen in Kombination mit einer partiellen Unvorhersehbarkeit des tatsächlichen Ablaufes dieser Prozesse impliziert eine gewisse Mindestanforderung an die Flexibilität dieser Dienstleistungsprozesse sowie den diesen zugrundeliegenden Modellen, vor allem hinsichtlich der Berücksichtigung relevanter Merkmale der jeweils gegebenen Situation beim Kunden. Auf diesen Kontextbezug von Dienstleistungen wird an späterer Stelle sowohl in anwendungsbezogener als auch in technischer Hinsicht eingegangen.

Der Ablauf von Dienstleistungen lässt sich drei in Tab. 11 dargestellte Phasen untergliedern, auf welche in den Kap. 3.7.2–3.7.4 genauer eingegangen wird. Eine Beschreibung weitergehender Eigenschaften des Dienstleistungssektors wie das *Uno-acto-Prinzip* finden sich in zusammengefasster Form in [Bode99].

Obwohl sie fast immer materielle Betriebsmittel beinhalten, haben Dienstleistungen einen prinzipiell *nichtmateriellen* Charakter; ihre Ein- und Ausgabeprodukte sind durch einen wesentlich höheren Anteil an Informationen gekennzeichnet als dies im primären oder sekundären Sektor der Volkswirtschaft üblich ist. Dieser starke Informationsbezug von Dienstleistungen ermöglicht in Kombination mit dem in den letzten Jahren erfolgten enormen Fortschritt im Bereich der Informations- und Kommunikationstechnik neuartige Dienstleistungsformen. An dieser Stelle werden Möglichkeiten, betriebliche Dienstleistungsvorgänge durch die explizite Berücksichtigung und Nutzung von Kontext zu unterstützen und zu erweitern, darstellt.

3.7.1 Allgemeine Verwendungsmöglichkeiten

Aus Kundensicht gilt es, Dienstleistungen möglichst *effektiv*, d. h. in einem möglichst umfangreichem Spektrum, und aus Anbietersicht möglichst *effizient*, d. h. gewinnmaximierend und damit indirekt kostengünstig zu erbringen. Ein dabei oft gewählter Ansatz ist die möglichst weitgehende Automatisierung wiederkehrender Prozesse, d. h. die Übertragung dieser Prozesse auf maschinelle Aufgabenträger, wodurch diese oft kostengünstig und in gleichbleibender und oft guter Qualität erbracht werden. Hochautomatisierte Dienstleistungen sind zwar machbar, können oft jedoch nicht in dem Maß wie bei menschlichen Bearbeitern auf die spezifischen Anforderungen des Kunden orientiert werden. Dies erschwert die Automatisierung von Dienstleistungsprozessen, d. h. die Substitution des Produktionsfaktors Arbeit durch Betriebsmittel. Durch den insgesamt höheren Automatisierungsgrad und die Kenntnis der Orte der beteiligten Subjekte kann nicht nur die Orts- und Zeitflexibilität, sondern zu einem gewissen Grad auch die *Situationsflexibilität* von Dienstleistungen erhöht werden, d. h. eine adaptierbare Dienstleistung kann dann auch in derart verschiedenen Situationen erbracht werden, die außerhalb der Einsatzmöglichkei-

Kriterium	Beschreibung
Nähe zum Kunden	Der Kunde möchte das Restaurant im Rahmen seines Stadtbummels zu Fuß erreichen.
Kulinarische Präferenzen des Kunden	Viele Nutzer dürfen aufgrund religiöser oder freiwilliger Einschränkungen bestimmte Lebensmittelarten, vor allem diverse Tier- bzw. Fleischarten nicht verzehren und legen daher Wert darauf, damit in Restaurants nicht konfrontiert zu werden, etwa europäische Touristen beim Bummel durch eine Gegend, in der auch Hundefleisch verzehrt wird.
Auslastung des Restaurants	Zu volle oder zu leere Restaurants sollen vermieden werden, sehr große Reisegruppen könnten dagegen gerade leere Restaurants bevorzugen, um entsprechend große zusammenhängende Sitzplatzmöglichkeiten zu haben oder zu lange Wartezeiten zu vermeiden.

Tabelle 12: Für das Matching beim Restaurantsuchprozeß relevante Kontextinformationen

ten von nichtadaptierbaren Dienstleistungen liegen. Im folgenden wird genauer auf die Einsatzmöglichkeiten von Kontextinformationen in den verschiedenen Dienstleistungsphasen eingegangen.

3.7.2 Verwendung von Kontext in der Anbahnungsphase

Die Anbahnungsphase (vgl. Tab. 11 („Kontextbezogene Dienstleistungsformen“, S. 93)) im Ablauf eines Dienstleistungsprozesses ist davon geprägt, daß die Anbieter ihre Dienstleistungen vorbereiten und mittels Marketing publizieren und die Nachfrager in diesen Publikationen einen geeigneten Anbieter suchen, der ihre Bedürfnisse befriedigen kann. Die Anbahnungsphase wird durch das *Matching*, d. h. das Finden eines geeigneten Geschäftspartners beendet.

Um einem Verbraucher bei der Suche nach einem passenden ad-doc-Dienstleister behilflich zu sein, ist es notwendig, dessen Situation, d. h. Kontext zu kennen. Zusätzlich sind viele Dienstleistungen hinsichtlich bestimmter Kontextparameter eingeschränkt, d. h. können nur an bestimmten Orten durchgeführt werden, welche i. d. R. mit dem gegenwärtigen Ort des Kunden zu vergleichen sind (vgl. [Mei⁺04]). Die automatisierte Verfügbarkeit von Kontextinformationen ermöglicht in diesem Zusammenhang zumindest eine teilautomatisierte Vorauswahl und Integration der zum aktuellen Kontext passenden Dienstleistungen.

In besonderem Maße gilt der Zusammenhang zwischen dem Kontext des Nachfragers und seinem Bedarf nach Dienstleistungen für *ad-hoc-Dienstleistungen*, worunter spontan und ungeplant nachgefragte Dienstleistungen verstanden werden. Der Bedarf für solche Dienstleistungen entsteht i. d. R. erst durch eine besondere, nicht vorhersehbare Situation und muß dann relativ zeitnah befriedigt werden. So hat jemand beispielsweise nur dann einen Bedarf nach einer Autoreparatur, wenn sein Auto defekt ist, und die Reparatur sollte möglichst am Ort des Schadensfalles stattfinden. Gleiches gilt für die Auswahl passender Restaurants in der Nähe des gegenwärtigen Standortes eines sich durch eine fremde Stadt bewegendem Touristen, wobei hinsichtlich der in Tab. 12 dargestellten Kriterien zu filtern ist.

3.7.3 Verwendung von Kontext in der Vereinbarungsphase

Nachdem sich in der Anbahnungsphase zwei Geschäftspartner, der Erbringer und der Verbraucher einer Dienstleistung, gefunden haben, werden in der daran anschließenden Vereinbarungsphase die konkreten Merkmale der zu erbringenden Dienstleistung spezifiziert, vor allem der Leistungsumfang, d. h. die einzelnen Leistungselemente und deren Ausgestaltung, die dafür zu erbringenden Vergütungen sowie der Ort und Zeitpunkt der Leistungserbringung.

Die konkrete Ausgestaltung einer Dienstleistung hängt neben den individuellen Präferenzen des Verbrauchers ebenfalls von der gegebenen Situation ab. Indem die relevanten Merkmale der gegebenen Situation elektronisch erfasst und verfügbar gemacht werden, können wesentliche Teile der Vereinbarungsphase teil- oder sogar vollautomatisiert werden, was neben der Möglichkeit, anbieterseitig Kosten zu sparen einen Zugewinn an Komfort für den Verbraucher bedeutet, da dieser letztlich nur an deren Ergebnis interessiert ist und sich nicht mit sekundären Tätigkeiten wie der Konfiguration einer Dienstleistung aufhalten möchte.

Beim Beispiel des havarierten KFZ ist es wichtig, wo genau sich das Fahrzeug befindet, um welchen KFZ-Typ es sich handelt sowie eine grobe Beschreibung des Schadensfalles, d. h. ob ein Personenschaden eingetreten ist, ob ein Ersatzfahrzeug benötigt wird oder ob etwa einfach nur der Treibstoff zur Neige gegangen ist. Ebenfalls relevant sind entsprechende Versicherungen des Nachfragers gegen solche Schadensfälle. Beim Beispiel des Restaurantbesuches gilt es, die individuellen Speisepreferenzen des Nachfragers zu berücksichtigen, etwa hinsichtlich des Verzichts auf bestimmten Lebensmittel, der Preisklasse des zu findenden Restaurants und des Aktivitätsradius des Verbrauchers.

3.7.4 Verwendung von Kontext in der Abwicklungsphase

In der Abwicklungsphase wird die Dienstleistung vom Dienstleister schlußendlich erbracht und entsprechend vom Nachfrager konsumiert. Kontextinformationen, d. h. Informationen über die gegebene Situation beim Verbraucher stellen auch hier eine wesentliche Grundlage für Entscheidungen, wie die Dienstleistung konkret zu erbringen ist, dar. Dies beginnt bei einer durch kontextsensitiv erfasste biometrische Merkmale wie etwa dem Gesicht von Personen gleichzeitig komfortabel, transparent und sicher gestalteten Identifikation von Personen. Basierend darauf kann auf ein öffentliches Nutzerprofil der betreffenden Person identifiziert und zugegriffen werden, wodurch – ergänzt durch Informationen über die gegenwärtige Situation dieser Person – die Dienstleistung an die individuellen Eigenschaften (Kenntnisstand, aktuelle Probleme) des Kunden angepasst werden kann.

3.8 Zusammenfassung

Während die unmittelbar für die betrieblichen Leistungsprozesse relevanten Informationen in Form von *Betriebsdaten* bereits seit geraumer Zeit maschinell erfasst werden, führt auch die Erfassung des Kontextes betrieblicher Prozesse zu einer Menge von *Kontextinformationen*, die vielfältig in betrieblichen Anwendungssystemen verwendet werden können, ebenso vielfältig sind die sich daraus ergebenden Effekte.

Neben einer Verbesserung der Qualität der verfügbaren Kontextinformationsbasis wird durch die Integration von heterogenen Kontextinformationen, welche aus diversen Datenquellen stammen und verschiedene Kontextdimensionen beschreiben, ein breites Spektrum von Kontextinformationen für sämtliche interessierten Anwendungen im betrieblichen Umfeld kostengünstig bereitgestellt.

Durch diese umfassenden Kontextinformationen werden nicht nur die durch betriebliche Prozesse zu verändernde Realweltdomäne, insbesondere die darin enthaltenen Einflußfaktoren auf das Verhalten von betrieblichen Anwendungen und Prozessen, sondern auch die realweltlichen Auswirkungen dieser Betriebsprozesse detailliert und umfassend beschrieben. Ein solches Kontextmodell führt neben der Möglichkeit für neuartige, kontextorientierte primäre Geschäftsmöglichkeiten also zu einer Reduzierung von non-value-added-activities, d. h. sekundären Betriebsprozessen, denn es stellt eine notwendige und geeignete Basis für die Analyse, Optimierung, Durchführung, Dokumentation und Bewertung einer breiten Palette betrieblicher Prozesse und damit einen wichtigen Schritt auf dem Weg zur möglichst vollwertigen Assistenz menschlicher durch maschinelle Aufgabenträger sowie auf dem Weg in Richtung der realen Utopie der Wirtschaftsinformatik, der sinnhaften Vollautomation des betrieblichen Geschehens, dar.

4 Verwandte Arbeiten

Für die Wirtschaftsinformatik und insbesondere für ein modernes Informationsmanagement war die Berücksichtigung der Aktivität des Empfängers für das Verständnis des Wesens der Information schon immer ein entscheidender Ausgangspunkt.

*Prof. Dr. Klaus Fuchs-Kittowski
Gesellschaft für Wissenschaftsforschung e.V.*

In diesem Kapitel werden existierende Arbeiten auf dem Gebiet der Modellierung von Kontext vorgestellt, wobei der Schwerpunkt auf Darstellungsmodelle gelegt wird, welche eine Integration heterogener Kontextinformationen erlauben. Diese werden hinsichtlich der in Kap. 2.4.1 („Bewertungsschema“, S. 25) entwickelten Bewertungskriterien auf ihre Tauglichkeit hinsichtlich der in dieser Arbeit untersuchten Problemstellung untersucht. Für ein besseres Verständnis dieser Domäne werden jedoch zunächst die wichtigsten innerhalb von Kontexterfassungssystemen verwendeten *low-level*-Kontextmodelle und danach existierende Ansätze für separate Teilbereiche der in dieser Arbeit entwickelten umfassenden Lösung vorgestellt.

4.1 Formen von Kontextvariablen

Generell beschreiben Kontexterfassungssysteme Realweltobjekte durch kontextbezogene Daten, die in dieser Arbeit *Kontextvariablen* genannt werden (vgl. Kap. 2.1.1 („Begriffsbestimmung“, S. 13)). Da Kontexterfassungssysteme oft auf die Erfassung und Darstellung spezifischer Kontextdimensionen spezialisiert sind, verwenden sie häufig auf diese Kontextdimensionen spezialisierte Kontextmodelle, von denen die wichtigsten im folgenden vorgestellt werden.

4.1.1 Stellvertreteransatz

Kontextinformationen werden in den einzelnen Kontextmodellen i. d. R. nicht isoliert dargestellt und organisiert, sondern einem Modell des von ihnen beschriebenen Realweltobjektes zugeordnet. Dieser sogenannte *Stellvertreteransatz* stellt ein grundlegendes Konzept zur Darstellung kontextbezogener Informationen dar und wird in den meisten, jedoch nicht allen Kontextmodellen angewendet. Stellvertreteransätze ermöglichen eine intuitive Darstellung der Realwelt sowie die einfache Integration von Kontextvariablen, indem diese einfach den betreffenden Realweltobjektmodellen zugeordnet werden, d. h. Realweltobjektmodelle im Realweltmodell übernehmen eine Vermittlerrolle zwischen den Realweltobjekten und den sie beschreibenden Kontextvariablen. Dieser Ansatz ist damit kompatibel zu dem von FERSTL und SINZ entwickelten *Objektsystem*, welches die Diskurswelt in Form von miteinander in Beziehungen stehenden Objekten modelliert (vgl. Kap. 2.1.2 („Das Unternehmens- und Umweltmodell in der Betriebswirtschaftslehre“, S. 15)).

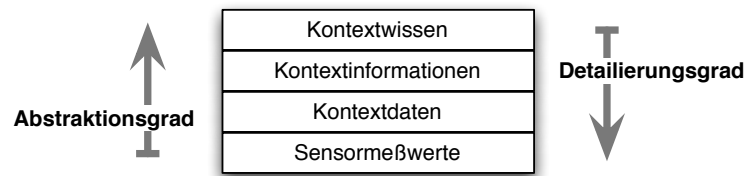


Abbildung 31: Kontextebenen

Die einfachste Realisierungsform dieses *Stellvertreteransatzes* stellen zu jedem Realweltobjekt zugeordnete elektronische Dokumente dar, ohne dass weitergehende Vorgaben über deren Struktur getroffen werden. Dieser Ansatz wurde im *Cooltown*-Projekt von BARTON und KINDBERG realisiert, bei dem Realweltobjekte mittels daran befestigter Sendeeinrichtungen eine elektronisch lesbare URL zu einem per Internet zugreifbaren Dokument versenden, welches dieses Objekt näher beschreibt (vgl. [BaKi01]). Weiterentwicklungen dieses Konzeptes führen über Strukturierungen dieser Informationsmengen (vgl. [Spr⁺01]) bis zu agentenbasierten Konzepten, bei denen Realweltobjekte nicht nur passiv beschrieben, sondern von einem aktiven Agenten repräsentiert und unterstützt werden, beispielsweise durch das Anbieten situationsbezogener Dienste bezüglich dieses Realweltobjektes. Dieser Ansatz wird u. a. vom *Mobile Shadow*-System von FISCHMEISTER realisiert, bei dem sich ein den Nutzer repräsentierender Agent in die Festnetzinfrastruktur des jeweils verfügbaren WLAN-Hotspots begibt und dort kontextbezogene Dienste anbietet (vgl. [Fisc03]).

4.1.2 Kontextebenenmodelle

Infrastrukturen zur Bereitstellung von Kontextinformationen werden i. d. R. in eine Schichtenarchitektur untergliedert, um die Funktionen zur Beschaffung von denen zur Auswertung der Kontextinformationen zu trennen und sämtliche dabei verwendeten Funktionseinheiten austauschbar sowie durch mehrere kontextsensitive Anwendungen verwendbar zu gestalten.

Für diese Schichtenarchitektur hat sich noch kein allgemein anerkanntes Schema durchgesetzt, jedoch können bei sämtlichen Ansätzen die in Abb. 31 dargestellten Schichten unterschieden werden. In diesen Schichtenarchitekturen werden auf der untersten Schicht die Sensoren und die von ihnen bereitgestellten *low-level-Kontextdaten* dargestellt. Diese Kontextdaten werden auf den mittleren Schichten stufenweise verdichtet, abstrahiert, transformiert und miteinander kombiniert, wodurch *Kontextinformationen* entstehen, welche auf den oberen Schichten wiederum durch Einbettung in einen anwendungsspezifisch handlungsentscheidenden Zusammenhang in kontextbezogene *Wissensbausteine* integriert werden (vgl. [Hen⁺02, Dey⁺99a, Sch⁺99a, Ail⁺02]).

Das *3-Ebenen Modell* des *Context Toolkit* von DEY und ABOWD formalisiert vor allem die unteren und mittleren Kontextebenen zur Sensorsteuerung, Interpretation und Integration von Kontextdaten. *Widgets* stellen dabei standardisierte Schnittstellen zu Kontextsensoren und -aktoren dar, die abstrahierend von konkreten Sensorzugriffsverfahren und -protokollen allgemeine Zugriffsmechanismen auf Sensordaten, vor allem Lese- und Schreibkommandos, bereitstellen. *Interpreter* verdichten und transformieren die Sensordaten zu höherwertigen Kontextinformationen, während *Aggregatoren* mehrere

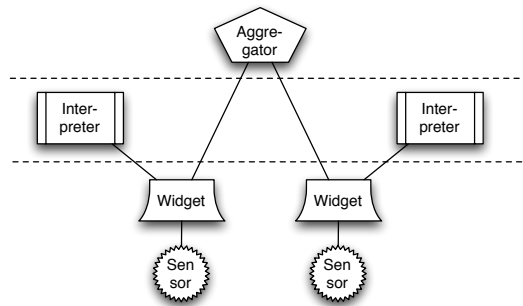


Abbildung 32: 3-Ebenen-Modell des *Context Toolkit*

zusammengehörige Kontextinformationen adressierungstechnisch zusammenfassen (vgl. Abb. 32, vgl. [Dey00b, S. 44 ff]).

Das von SCHMIDT im *TEA*-Projekt verwendete 4-Ebenen-Modell weist neben den 3 Ebenen des *Context Toolkit* bereits einen Situations- und Anwendungsbezug auf, indem die Wahrscheinlichkeit für die gegenwärtige Gegebenheit typischer Situationen erfragt sowie kontextbezogene Trigger, die beim Eintreten, während bzw. beim Verlassen einer bestimmten Situation ausgelöst werden, gesetzt werden können (vgl. Abb. 33 auf S. 99, [Sch⁺99a, S. 101]).

Einen analogen Ansatz, jedoch mit 5 Ebenen, verwendet AILLISTO, wobei die einzelnen Kontextinformationen hier im Gegensatz zu SCHMIDT nicht zu Situationen, sondern wie allgemein üblich zu Realweltobjektrepräsentationen zusammengefasst werden. Anwendungsspezifisch können hier ebenso wie bei SCHMIDT nur kontextbedingte Verhaltensweisen, jedoch keine Sichtweisen auf die Datenbasis definiert werden (vgl. [Ail⁺02]).

4.1.3 Räumliche Weltmodelle

Die am häufigsten von kontextsensitiven Anwendungssystemen berücksichtigte Kontextdimension stellt der physikalische Ort sowie damit zusammenhängende Kontextvariablen wie die Geschwindigkeit, Beschleunigung oder der Weg eines Kontextobjektes dar. Ortsbezogene Kontextmodelle umfassen nicht nur die Identifikation der Positionen von Objekten, sondern auch die Beschreibung dieser Orte. Dabei können zwei grundlegende Arten von räumlichen Weltmodellen unterschieden werden, welche im folgenden konkreter vorgestellt werden.

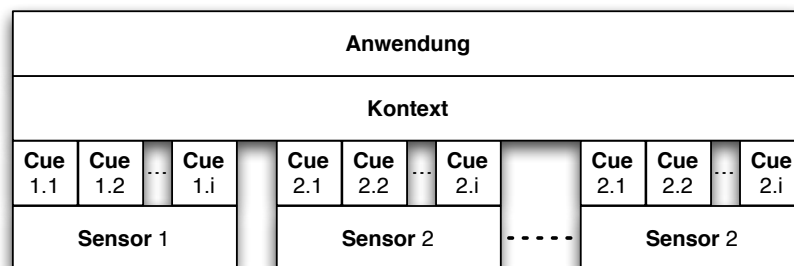


Abbildung 33: Organisation von Kontextvariablen im *TEA*-Modell

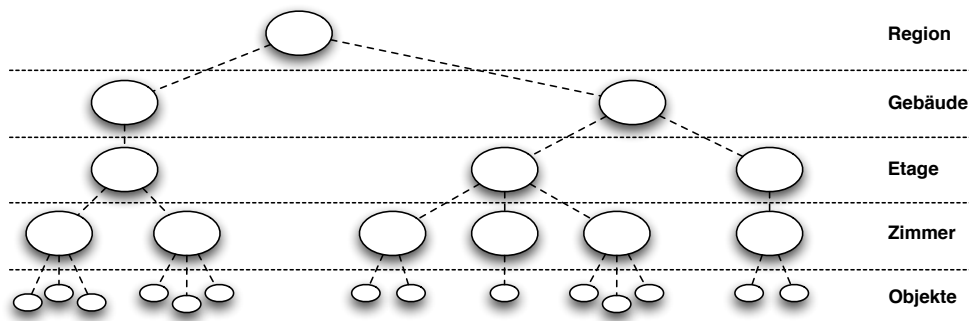


Abbildung 34: Aufbau einer Active Map

4.1.3.1 Symbolische Ortsmodelle

Symbolische Modelle modellieren die logischen Beziehungen verschiedener Positionsangaben, etwa die Information, dass sich ein bestimmtes Zimmer in einem bestimmten Gebäude und dieses wiederum auf einem Campus befindet und können daher vor allem die Beziehungen zwischen Orten gut beschreiben (vgl. [ScTh94, Leo⁺96]).

Active Maps stellen ein symbolisches Weltmodell dar, das aus hierarchisch organisierten, aus Attribut-Wert-Paaren bestehenden Beschreibungen von Orten bzw. Positionen besteht (vgl. Abb. 34). Dabei sind Ortsangaben auf unteren Ebenen in den Orten der jeweils oberen Ebene enthalten. Dadurch wird es möglich, sowohl grobe als auch feingranulare Ortsangaben im selben Modell darzustellen und Informationen zu jeder Granularitätsstufe individuell anzugeben.

4.1.3.2 Geometrische Ortsmodelle

Geometrische Modelle stellen räumliche Positionen von Realweltobjekten mittels skalarer physikalischer Meßgrößen, z. B. GPS-Koordinaten oder Längen- und Breitengraden geometrisch korrekt dar und können daher Ortsarithmetiken, etwa die Berechnung von Entfernungen zwischen Positionen, geometrisch korrekt realisieren (vgl. [Har⁺01]).

Beim geometrischen *Worldboard*-Modell wird die Realwelt in Kubikmeter und -zentimeter, sogenannte *Cubes*, unterteilt, wobei jedem Cube beliebige Informationen angehängt werden können (vgl. [Spoh99]). Mit diesem Ansatz lassen sich vor allem immobile Objekte beschreiben. Ein gleicher Ansatz stellt der an der Universität von Stuttgart entwickelte NEXUS dar (vgl. [Vol⁺99]).

Analoges gilt für den *Stick-e-notes*-Ansatz, bei dem eine bestimmte Kontextsituation durch elektronische Notizen beschrieben werden kann. Eine *Kontextsituation* wird durch beliebige Kombinationen von Kontextparametern adressiert, wobei den räumlichen und zeitlichen Kontextdimensionen jedoch eine dominante Rolle zufällt. Dieses Modell wurde erfolgreich in einer Reihe von mobilen Beobachtungsanwendungen etwa zur Tierbeobachtung in der freien Natur eingesetzt, hat jedoch auch einen eher statischen Charakter (vgl. [Pasc97, Pas⁺98, Pas⁺99]).

Um den Kontext mobiler Objekte zu beschreiben, wurde im *Spatially Situated Information Spaces*-Projekt ein Kontextmodell entwickelt, bei dem Informationen über ein (mobiles) Realweltobjekt dem

dreidimensionalen Raum um den entsprechenden physikalischen Körper herum zugeordnet werden (vgl. [Fitz93]).

4.1.4 Semantische Weltmodelle

Im Gegensatz zu den räumlichen Weltmodellen, die den räumlichen Aufbau der Realwelt modellieren, stellen semantische Weltmodelle realweltobjektbezogene Zusammenhänge beliebiger Natur dar und erzeugen dadurch eine flexibler recherchierbare Topologie von Kontextinformationen, die zur Beantwortung einer größeren Palette von Abfragen verwendet werden kann.

Semantische Weltmodelle bestehen i. d. R. aus einer Menge von Realweltobjektrepräsentationen, die miteinander durch semantisch bedeutsame Verbindungen assoziiert sind. Wegen ihrer problemübergreifenden Verwendbarkeit werden semantische Weltmodelle von der Mehrheit der kontextverarbeitenden Systeme verwendet. Daher spielen sie im weiteren Verlauf der Arbeit eine dominante Rolle, d. h. sämtliche im folgenden vorgestellten Weltmodelle stellen semantische Weltmodelle dar.

4.1.5 Situationsmodelle

Während *Kontextsensitivität* den Oberbegriff für sämtliche Verfahren zur Erfassung realweltbezogener Parameter darstellt und sämtliche zu Realweltobjekten zuordbaren Informationen und Wissenbestandteile, etwa die Dokumente eines Nutzers modelliert, wird unter *Situationssensitivität* die Verwendung von Kontextinformationen in konkreten Situationen, d. h. hinsichtlich tatsächlich zu einem konkreten Zeitpunkt auftretender Probleme, verstanden (vgl. [ScGe01]).

Eine *Situation* stellt in diesem Zusammenhang den augenblicklichen Zustand, d. h. eine Momentaufnahme eines aufgabenbezogenen Realweltausschnittes dar (vgl. u. a. [McHa68]). Sie wird beschrieben durch den aktuellen Wert aller diesbezüglich relevanten variablen Kontextelemente. Neben den Kontextvariableninhalten interessieren bei situationsbewussten Anwendungen jedoch vor allem Schlußfolgerungen über die aktuelle Situation. Diese Situationsparameter werden häufig in den i. d. R. sehr komplexen kontextbezogenen Entscheidungsprozessen benötigt und stellen daher die Grundlage für eine Reihe weiterer zeit- und ablaufbezogener Überlegungen dar, etwa warum gewisse Dinge gerade passieren, welche Prozesse danach folgen werden und wie darauf reagiert werden kann (vgl. [Dey00b, S. 100 ff], [Dey01a], [Mei⁺04, S. 20]). Beispiele für situationsbezogene Angaben stellen die Aussagen „*der Nutzer steuert gerade ein Kraftfahrzeug*“ oder „*Das KFZ ist gerade in einem Parkhaus abgestellt*“ dar. Im folgenden werden die wichtigsten innerhalb von Kontextquellen verwendeten Ansätze zur Darstellung von Situationen vorgestellt.

4.1.5.1 Context Schemas

Die von TURNER basierend auf fallbasierten Entscheidungsverfahren entwickelten *Context Schemas* beschreiben Erkennungsmerkmale für Situationen in einer assoziativen Form (vgl. [TuTu91, Turn92]). Dabei werden Situationen durch die Angabe des für diese Situation zulässigen Wertebereiches einzelner Kontextvariablen beschrieben. Eine Situation wird dann als gegeben angenommen, wenn sämtliche aktuellen Kontextvariablen Werte innerhalb der beschriebenen Bereiche aufweisen. Für jede Situation

<p>Merkmale: Wassertiefe: flach Geräuschpegel: hoch Schiffsverkehr: hoch</p> <p>Anweisungen: setze maximale Tauchtiefe setze minimale Tauchtiefe Hindernisausweichmodus an</p>	<p>Verhaltensweisen: Systemversagen: auf Grund landen, Boje starten Hindernis: zuerst versuchen, darunter zu tauchen</p> <p>Verfahrensweisen: Lokalisierung: benutze Karte + Beacons im Hafen</p>
--	--

Abbildung 35: Ausschnitt aus einem Context Schema für den Aufenthalt eines U-Bootes im Hafen

können sofort auszuführende Schritte, Richtlinien zur Verhaltensweise, etwa zur Reaktion auf verschiedene Ereignisse sowie konkrete Handlungsweisen, d. h. Anleitungen, wie einzelne Handlungen in dieser Situation auszuführen sind, gegeben werden. Ein Beispiel für eine Situationsbeschreibung, welche die Situation, dass sich ein U-Boot in einem Hafen befindet beschreibt und dafür geltende Verhaltensrichtlinien festlegt, ist in Abb. 35 gegeben (vgl. [Turn92, S. 3]).

4.1.5.2 Information Logistics

Im *Message Logistics*-Projekt, das auf dem *Information Logistics*-Ansatz des Fraunhofer Instituts für Software und Systemtechnik basiert, werden elektronische Nachrichten kontext- und situationsbezogen an mobile Nutzer ausgeliefert. Dazu werden die Themen, zu denen eine Nachricht Aussagen macht, bestimmt und analysiert, ob diese Themen für den Nutzer in seiner aktuellen Situation relevant sind. Beispielsweise ist die Nachricht, dass der gerade in der Werkstatt befindliche PKW des Nutzers bereit zur Abholung ist, innerhalb eines Meetings nicht relevant und wird deshalb während des Meetings nicht zugestellt.

Information Logistics beschreibt Situationen ebenfalls durch eine Reihe von Kontextvariablen, wobei auch deren zeitliche Abfolge berücksichtigt wird. Dadurch lassen sich Sequenzen von Situationen darstellen, die wiederum Schlußfolgerungen über mögliche zukünftige Situationen und den dann entstehenden Informationsbedarf zulassen, was die Grundlage für proaktive Vorbereitungen darstellt. Ein Beispiel stellt die Schlußfolgerung dar, dass ein Nutzer in Zukunft auf einem Flughafen sein wird, wenn er jetzt in einem Flugzeug sitzt. Zur Bestimmung, ob eine Nachricht an den Nutzer zugestellt werden soll, wird deren Relevanz in der gegebenen Situation sowie ihr Nutzenpotential hinsichtlich der aktuellen Probleme des Nutzers bestimmt und durch geeignete Akzeptanz- und Nutzenfunktionen dargestellt (vgl. [Jak⁺03, Mei⁺04]).

4.1.5.3 TEA

Das *TEA*-Projekt interpretiert Messwerte, im *TEA*-Modell *Cue* genannt, zwar durchaus als Attribute von Realweltobjektpräsentationen, berücksichtigt diese Realweltobjektmodelle jedoch nicht weiter sondern fasst die Messwerte modellintern zu Situationsmodellen zusammen und stellt daher vor allem situationsbezogene Primitive für Clientanwendungen bereit (vgl. Abb. 33 auf Seite 99). Dazu gehören

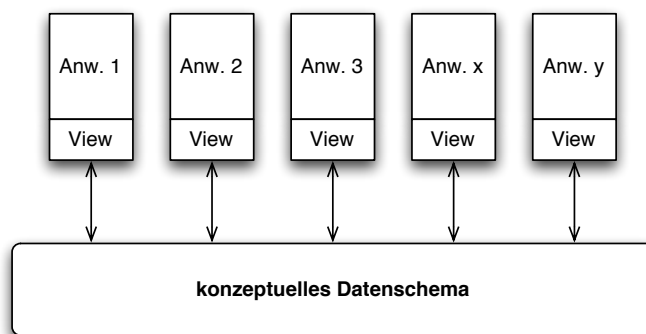


Abbildung 36: Datenintegration

Benachrichtigungen, sobald eine Situation eintritt, wenn sie beendet wird bzw. periodisch während der Dauer einer Situation (vgl. [Sch⁺99a, S. 101 f]).

Eine *Situation* wird durch den Wertebereich beschrieben, den bestimmte Kontextvariablen innerhalb dieser Situation annehmen dürfen. Diese Angabe erfolgt durch Fuzzy-Funktionen, d. h. Funktionen, welche die Zugehörigkeit konkreter Messwerte zur gesuchten Menge durch eine Wahrscheinlichkeit angeben (vgl. [Zade73], für mehr Informationen siehe [Trae94]). Die Verfügbarkeit konkreter Sensormesswerte erlaubt nun, festzustellen, welche Situationen mit welcher Wahrscheinlichkeit zum aktuellen Zeitpunkt als gegeben anzunehmen sind.

4.2 Integration heterogener Informationsmengen

Nachdem die wichtigsten Formen von innerhalb von Kontextquellen verwendeten Datenstrukturen vorgestellt wurden, wird an dieser Stelle auf das Problem der Integration von heterogenen Informationen in eine alles umfassende Sichtweise eingegangen. Dabei werden zunächst die diesbezüglichen Instrumente aus der klassischen Datenhaltung und anschließend auf die Integration von Kontextinformationen spezialisierten Verfahren vorgestellt und hinsichtlich ihrer Tauglichkeit für die gegebene Aufgabenstellung bewertet.

4.2.1 Unternehmensweite Datenmodelle

Mit Einführung von Datenbanksystemen und Mehrplatz-Dialogsystemen in den 1970er und 1980er Jahren wurden datenbankbasierte Integrationskonzepte von betrieblichen Anwendungssystemen diskutiert und eingesetzt (vgl. [Sche78, Müll89, Mei⁺91, Hanf91]). Dabei wird die gesamte Datenstruktur eines Unternehmens vom konzeptuellen Datenschema eines (i. d. R. relationalen) Datenbanksystems repräsentiert. Für jedes Teilsystem werden Teilmengen dieses Datenschemas als individuelle *Sichten* (Views) auf die Datenstruktur bereitgestellt (vgl. [FeSi98], Abb. 36).

Klassische datenbankgestützte Integrationskonzepte sind auf die Optimierungsziele *Redundanzverringern* und *Erhalt der Integrität* ausgerichtet. Der Einsatz der in diesem Zusammenhang notwendigen *unternehmensweiten Datenmodelle* (UDM) konnte in der Praxis jedoch nicht ohne Probleme erfolgen, da deren Erstellung und Pflege sehr aufwändig ist. Problematisch ist in diesem Zusammenhang die trotz

bereits strukturierter und dokumentierter Abläufe und Datenschemata in den einzelnen Abteilungen vorhandene Schwierigkeit, die Komplexität des entstehenden unternehmensweiten Datenmodells sowie die in diesem Zusammenhang notwendigen Abstimmungsprozesse zwischen allen Unternehmensteilen z. B. hinsichtlich einer einheitlich zu verwendenden Terminologie zu handhaben (vgl. u. a. [FeSi98, S. 230]).

Insbesondere die Komplexität des gesamten Datenmodells wird bei der Verarbeitung von Kontextinformationen jedoch nochmals verschärft, da Kontextdaten i. d. R. mit einer höheren Detailierung modelliert werden, da sie nicht bewusst *abstrahieren*, sondern möglichst *konkrete*, fallbezogene Aussagen über die jeweilige Situation treffen sollen. Zudem treten kontextbezogene Daten nicht immer in der benötigten stark strukturierten Form sowie in sehr heterogenen Ausprägungen und Formaten auf, was die Erstellung eines einheitlichen Datenmodells erschwert. Die nur teilweise Verfügbarkeit, die nicht unbedingt vorhandene Vertrauenswürdigkeit bzw. Korrektheit sowie durch ungenaue Modellierung entstehende Widersprüchlichkeit von kontextbezogenen Informationen macht die Verarbeitung dieser Art von Informationen in für den beschriebenen Unternehmenseinsatz konzipierten Datenbanken unmöglich. Fernerhin sind die Verarbeitungsrichtlinien bei Kontextinformationen und Betriebsdaten teilweise gegensätzlich. So sollen Unternehmensdaten i. d. R. transaktionsorientiert und sicher verarbeitet werden, während der Schwerpunkt bei der Verarbeitung von Kontextinformationen auf deren möglichst performanten Verarbeitung liegt. Aus diesen Gründen kann bei typischen kontextbezogenen Aufgabenstellungen keine einheitlich strukturierte unternehmensweite Kontextdatenstruktur vorausgesetzt bzw. eingesetzt werden. Diese stellt jedoch eine Voraussetzung für das dargestellte datenbankgestützte Integrationskonzept dar.

Gleiches gilt für die moderneren *Enterprise Application Frameworks*, welche flexible Architekturen darstellen, mittels welchen alle Arten von kooperierenden Unternehmensanwendungen realisiert werden können. In der Regel existieren in solchen Infrastrukturen zahlreiche Möglichkeiten, um externe Anwendungskomponenten und somit beliebige Dienste einzubinden und so auch fortgeschrittene Infrastrukturkonzepte wie verteilte oder mobile Anwendungen mit allen notwendigen Sicherheitskonzepten zu ermöglichen. Eine Kontextinformationsbasis wäre in einem solchen Szenario eine weitere Komponente, die von allen beteiligten Anwendungen nach Bedarf verwendet werden kann, um ein kontextbezogenes Verhalten zu realisieren. Allerdings weisen auch solche Systeme i. d. R. nicht die für die Verarbeitung von Kontextinformationen benötigte Echtzeitfähigkeit auf (vgl. [Gos⁺03, S. 3]).

4.2.2 Enterprise Application Integration

Zur Einbindung veralteter Unternehmensanwendungen, die keine Interaktion mit anderen Anwendungen vorsehen und nicht im Rahmen einer Migrationsstrategie durch modernere, flexiblere Systeme ersetzt werden sollen oder können, was an zu hohen Migrationskosten oder einer Nichtverfügbarkeit moderner Nachfolgesysteme liegen kann, werden Verfahren der *Enterprise Application Integration* (EAI) verwendet. EAI stellt dabei einen Sammelbegriff für Konzepte, Methoden und Softwarewerkzeuge zur Integration und Koordination bislang isolierter Anwendungssysteme *innerhalb* von Unternehmen dar. Eine firmenübergreifende Anwendungsintegration wird dagegen als *Business Integration* bezeichnet. Ziel von EAI ist die Realisierung systemübergreifender Geschäftsprozesse mit den in einem Unternehmen bestehenden (Legacy)-Anwendungen, die dies eigentlich nicht vorsehen. Dazu transformieren sie die zwi-

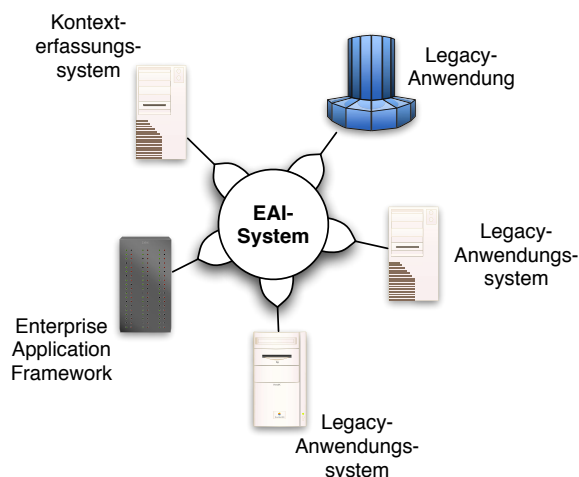


Abbildung 37: Typische Topologie eines EAI-Systems

schen den Anwendungen ausgetauschten Informationseinheiten in die für jeden Kommunikationspartner passende Form und ermöglichen dadurch eine Kommunikation auch zwischen heterogenen, nicht direkt kompatiblen Unternehmensanwendungen, wobei sichergestellt wird, dass alle beteiligten Interaktionspartner das selbe Verständnis der ausgetauschten Informationen haben (vgl. [AuFr01]). Allerdings haben EAI-Systeme aufgrund der Komplexität der Domäne noch nicht die in sie gesetzten Erwartungen bzw. die Versprechungen ihrer Hersteller erfüllen können.

EAI-Systeme bestehen i. d. R. aus einer *Hub-and-Spoke* genannten Architektur, welche eine *Hub* genannt zentrale Austauschplattform beinhaltet, die auszutauschende Nachrichten transformiert, mit fehlenden Daten anreichert sowie mittels *Spoke* genannten anwendungsspezifischer Konnektoren zu den entsprechenden Zielanwendungen weiterleitet (vgl. Abb. 37). Da die beteiligten Legacy-Anwendungen über unterschiedliche Arten von Nachrichten kommunizieren, sind dabei u. U. komplexe Transformationen innerhalb der zentralen Hub-Komponente notwendig.

Da EAI-Systeme die notwendige Infrastruktur zum Zugriff auf Legacy-Anwendungssysteme bereitstellen, können sie sowohl zur Beschaffung als auch zur Anwendung von kontextbezogenen Informationen verwendet werden, denn die durch EAI-Systeme ermöglichten systemübergreifenden Geschäftsprozesse können durchaus auch Kontextinformationen berücksichtigen. Durch eine kontextbezogene Steuerung durch das EAI-System könnte auch in eigentlich nicht kontextsensitiven Unternehmensanwendungen ein kontextbezogenes Verhalten erreicht werden. Innerhalb einer solchen kontextbezogenen EAI-Lösung könnte das in dieser Arbeit entwickelte Darstellungsmodell für integrierte kontextbezogene Informationen verwendet werden. Diesbezügliche Untersuchungen stehen noch aus und müssen auch im Rahmen dieser Arbeit in den Ausblick verschoben werden.

4.2.3 Operational Data Store

Ein *Operational Data Store* (ODS) sammelt genau wie ein *Data Warehouse* (DWH) operative Anwendungsdaten in einem anwendungsübergreifenden, realweltobjektorientierten¹⁷ Datenmodell, allerdings

¹⁷INMON spricht von *subjektorientierten* Datenmodellen, worunter jedoch das gleiche zu verstehen ist (vgl. [Inmo99, S. 13]).

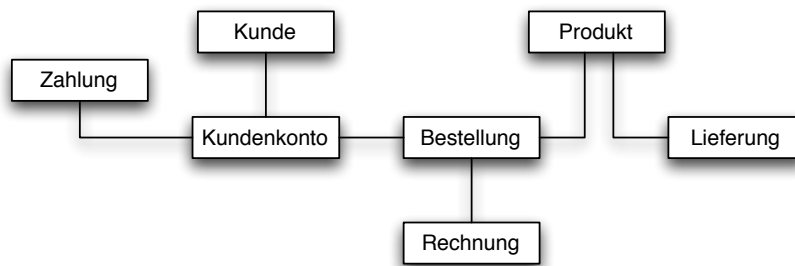


Abbildung 38: Beispiel für ein typisches in einem ODS verwendetes Datenschema (vgl. [Inmo99, S. 86])

sollen diese bei einem ODS im Gegensatz zu einem DWH auch wieder operativ verwendet werden, weshalb ODS i. d. R. thematisch abgegrenzter und damit performanter als DWH konzipiert werden. Wie beim Data Warehousing ist ein ODS nicht direkt mit den beteiligten Anwendungen, sondern mittels einer Integrationsplattform mit deren Datenbanken verbunden und vollzieht die dort durchgeführten Transaktionen zeitnah nach (vgl. [Inmo95, S. 13]). Da durch ein ODS sämtliche beteiligten Anwendungen auf dem gleichen Datenmodell operieren, können operative Anwendungsdaten zwischen heterogenen Anwendungen ohne direkte Kopplung der Anwendungssysteme ausgetauscht und miteinander integriert werden.

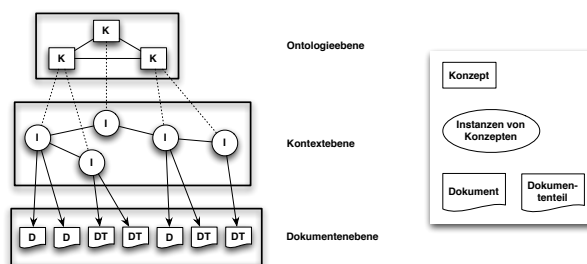
Bezüglich der Integration kontextbezogener Informationen kann festgestellt werden, dass ODS eine Infrastruktur zur echtzeitfähigen Abbildung der operativen Datenbasis eines Unternehmens und damit ein Mittel zur Integration operativer Daten in verschiedene Unternehmensanwendungen darstellen. Dies kann prinzipiell auch Kontextinformationen einbeziehen. Allerdings sind die bei ODS verwendeten ERM und Star-Schemata auf eindeutige, widerspruchsfreie und vollständig modellierte Informationen ausgelegt und damit nur bedingt zur Verarbeitung von Kontextdaten geeignet (vgl. Abb. 38). Darüber hinaus liegen Kontextinformationen im Gegensatz zu Unternehmensdaten nicht zwangsläufig im relationalen Datenmodell vor und müssen daher aufwändig bzw. mittels Transformationsverlusten in dieses recht einfache Modell transformiert werden. Zudem können die von ODS verwendeten Datenschemata semantische Informationen über die enthaltenen Informationen nur unzureichend darstellen und sind daher in semantischer Hinsicht nur bedingt aussagefähig.

Eine kontextbezogene Informationsintegrationsarchitektur könnte also zumindest basierend auf der Architektur eines normalen ODS und unter Verwendung des in dieser Arbeit entwickelten Darstellungs- und Integrationsmodells für kontextbezogene Informationen realisiert werden.

4.2.4 Semantic Web

The Semantic Web is a vision: the idea of having data on the web defined and linked in such a way that it can be used by machines - not just for display purposes, but for using it in various applications. Tim Berners-Lee

Die von BERNERS-LEE vorgeschlagene Idee einer *Semantic Web* genannten Weiterentwicklung des klassischen World Wide Web in Form einer zusätzlichen, semantisch aussagefähigen Ebene im WWW, welche die in Webdokumenten enthaltenen Informationen in einer auch für maschinen interpretierbaren Form beschreibt, ermöglicht eine zumindest in abgegrenzten Domänen automatisierte Auswertung und

Abbildung 39: 3-Ebenen-Struktur des *Semantic Web*

Integration des Inhaltes von Webseiten durch Computeranwendungen (vgl. [Bern98, Ber⁺01, ScGe03, Hefl01]).

Um die Inhalte von Webseiten maschinenverständlich aussagefähig zu gestalten, werden sie mittels der in Abb. 39 dargestellten dreistufigen Hierarchie formalisiert (vgl. [ScGe03, S. 6]). Dabei werden die betreffenden Inhalte, deren Verhältnis zueinander sowie das dazu notwendige Bedeutungswissen voneinander getrennt auf verschiedenen Ebenen modelliert. Auf der *Dokumentenebene* werden die jeweiligen Inhalte in Form beliebiger Webdokumente, d. h. webzugänglicher Dokumente in textueller, multimedialer oder in Form strukturierter Daten vorgehalten. Dieser z. T. bereits heute existierende Webdokumentenbestand wird nun durch eine separate *Kontextebene* semantisch beschrieben und in Beziehung zueinander gesetzt. Dabei werden einzelne Dokumententeile auf der Dokumentenebene zu semantisch determinierten Konzepten auf der Kontextebene zugeordnet. Dabei wird üblicherweise der in Kap. 4.1.1 auf S. 97 vorgestellte *Stellvertreteransatz* gewählt, d. h. die einzelnen Webdokumente werden nicht direkt einander, sondern expliziten Realweltobjektrepräsentationen zugeordnet. Da diese Zuordnungen typisiert sind, können einzelne Elemente der Dokumentenebene verschiedenen Aspekten eines Realweltobjektes zugeordnet werden. Diese Realweltobjektrepräsentationen sind wiederum untereinander durch typisierte Assoziationen verbunden, die im Gegensatz zu den im heutigen WWW üblichen untypisierten *Hyperlinks* bedeutungsmäßig determiniert sind. Das hierfür benötigte semantische Bedeutungswissen wird auf einer expliziten *Ontologieebene* durch Ontologien definiert (vgl. [Schw02, S. 3]). Auf diese Weise können eigentlich voneinander isolierte Informationsfragmente aus verschiedenen Dokumenten durch die übergeordnete Kontextebene semantisch miteinander in einen Zusammenhang gebracht und damit *dokumentenübergreifendes Wissen* dargestellt werden.

Beispiel 12 *Ein webzugängliches Bild ist als Darstellung einer bestimmten Person gekennzeichnet, welche wiederum als CEO eines gewissen Unternehmens beschrieben wird. Eine in einem anderen Dokument auftauchende Zahl wird als Umsatz dieses Unternehmens gekennzeichnet. Durch diese Verbindungen einzelner Dokumententeile auf der Kontextebene werden Anfragen der Form „Zeige alle CEOs von Firmen, die KFZ-Teile herstellen und über 15.000.000 Euro Umsatz im Jahr generieren“ basierend auf einer Recherche des WWW möglich.*

Das *Semantic Web* stellt geeignete Methoden zur Integration heterogener webbasierter Informationsmengen zu dokumentenübergreifenden Wissensstrukturen bereit. Es eignet sich prinzipiell zur Integration sämtlicher Informationsarten, ist jedoch nicht im speziellen auf die besonderen Anforderungen

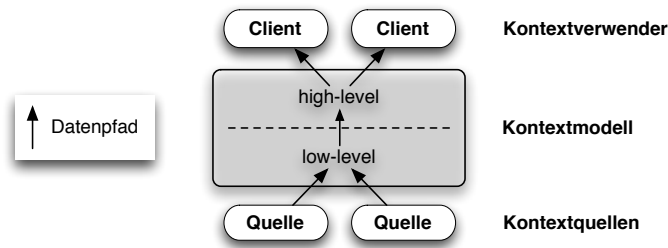


Abbildung 40: 3 Ebenen der Kontextdatenverarbeitung

von Kontextinformation ausgelegt. Diese Aussage wird in Kap. 4.5.4 und 4.5.5 durch eine detailliertere Bewertung der beiden wichtigsten Realisierungsformen des *Semantic Web*, RDF und Topic Maps, hinsichtlich ihrer Eignung zur Integration kontextbezogener Informationen untermauert.

4.3 Integration aktiver Elemente in Datenstrukturen

Nur wenige Darstellungs- und Integrationsmodelle für Informationen adressieren die in Anf. 6h („Integration von Methodik in das Datenmodell“) geforderte Integration von algorithmischer Logik in das Kontextdatenmodell. Dabei geht es um eine quasi „selbsttätig“ vom Kontextmodell realisierte Verarbeitung, d. h. Auswahl, Transformation und Interpretation der modellinternen *low-level*-Informationen in eine höherwertigere, d. h. *high-level*-Form (vgl. Abb. 40). Im folgenden werden die wichtigsten Ansätze dazu vorgestellt und bewertet.

4.3.1 Externe Agenten

Häufig werden zur Informationstransformation *Agenten* genannte externe Instanzen eingesetzt, welche einzelne Informationselemente in der Kontextdatenbasis überwachen, interpretieren und ihre ggf. geschlußfolgerten Informationen an anderer Stelle zur Datenbasis hinzufügen (vgl. Abb. 41). Dieser Ansatz wird u. a. in den später noch konkreter vorgestellten Projekten *Context Shadow* und *Context Toolkit* verwendet.

Problematisch an diesem Ansatz ist, dass durch die Trennung von Daten und zu diesen Daten gehöriger Methodik den Informationen in der Kontextdatenbasis nicht angesehen werden kann, wann sie wie von welchem Agenten verarbeitet werden. Als Ergebnis kann sich kein wirklich zusammenhängendes

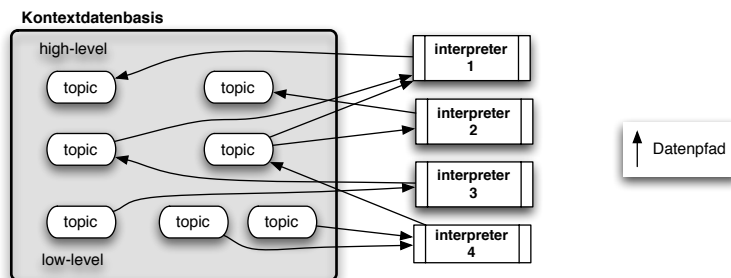


Abbildung 41: Interpretation von Kontextinformationen durch externe Agenten

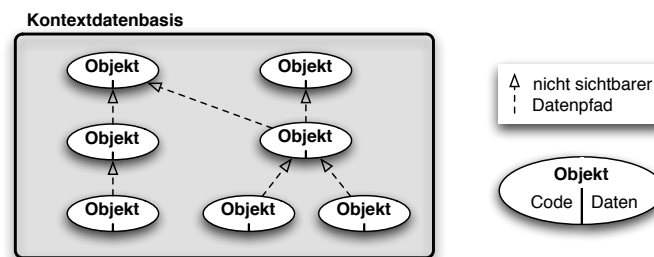


Abbildung 42: Kapselung der Interpreter in Objektstrukturen

Gesamtbild der Kontextwissensbasis in komplexen Szenarien ergeben. Widersprüchliche Verarbeitungen oder Schleifen im Verarbeitungsmodell können durch die Verteilung der entsprechenden Methodik auf mehrere Agenten und den dadurch nicht sichtbaren Verarbeitungsweg der Informationen schwer entdeckt werden (vgl. Anf. 6i („Darstellung des Datenpfades“)). Fernerhin kann durch den separaten Überwachungsvorgang durch einen externen Agenten nicht immer sichergestellt werden, dass die betreffenden Daten *sofort* verarbeitet werden, wodurch die Datenbasis u. U. veraltete und damit widersprüchliche Informationen enthalten kann. In Abhängigkeit von der Speicherungsart der Interpreter ist diese Auswertungsform auch nur bedingt replikationsfähig.

4.3.2 Objektorientierung

Eleganter als die Nutzung von externen Interpreteragenten ist der Ansatz, die datenbezogene Methodik zusammen mit den entsprechenden Daten darzustellen, wie dies beispielsweise im objektorientierten Datenmodell oder etwa bei den von FITZPATRICK entwickelten *Sentient Objects* der Fall ist, auf die in Kap. 4.6.6 („Sentient Computing“, S. 125) genauer eingegangen wird. Hierbei werden die Daten zusammen mit den entsprechenden Transformationsmethoden zu *Objekten* zusammengefasst, welche selbstständig *low-level*-Daten beschaffen und die daraus abzuleitenden *high-level*-Informationen bereitstellen können (vgl. Abb. 42).

Durch den objektorientierten Ansatz können Berechnungen transparent in die Daten- bzw. Objektstruktur integriert werden, wobei auf eine Reihe von effizienten Modellierungstechniken wie Vererbung zurückgegriffen werden kann. Objektorientierung ist jedoch primär ein programmiertechnisches Paradigma und nicht direkt zur Integration von Transformationsmethodik in Datenstrukturen konzipiert. Durch die bei Objektstrukturen sehr enge Verbindung von Daten und Code sind die einzelnen Schritte der Verarbeitungskette von Kontextinformationen sowie deren Semantik nicht direkt sichtbar, da die Tatsache, das und wie Kontextinformationen miteinander kombiniert werden, nur implizit innerhalb eines Methodencodes ohne eine semantische Auszeichnung der Operatoren dargestellt werden kann. Im Modell ist daher nicht direkt sichtbar, welche Kontextinformationen mittels welcher Operationen aus welchen anderen Informationen bestimmt werden. Auch ist eine objektorientierte Datenstruktur durch ihren codeorientierten Charakter nur bedingt für Endbenutzer visualisierbar und auch nicht ohne weiteres zur Laufzeit veränderbar, insbesondere da hierdurch Anwendungscode und kontextverarbeitende Strukturen relativ eng miteinander verschmolzen werden. Durch diese enge Verschmelzung von Daten und Code zu Objektstrukturen geht außerdem der datenbezogene Charakter der Datenbasis verlo-

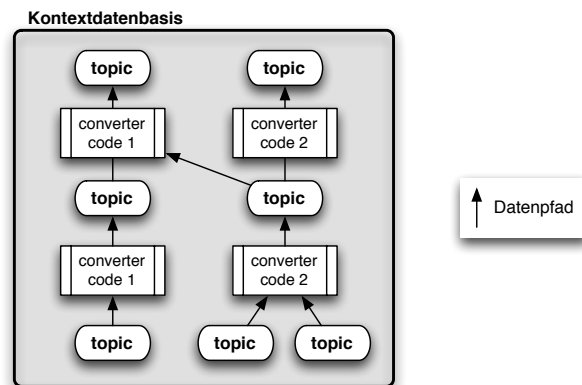


Abbildung 43: Integration von Interpretern in die Datenstruktur mittels *aktiver Assoziationen*

ren, wodurch der Einsatz modellexterner Auswertungsmechanismen, etwa von Expertensystemen oder Data-Mining-Verfahren, problematisch wird, da diese i. d. R. nur Daten-, aber keine Objektstrukturen auswerten können.

4.3.3 Aktive Assoziationen

Die eben dargestellten Nachteile objektorientierter Verfahren werden durch *Aktive Assoziationen* adressiert. Diese modellieren nicht nur die semantischen Zusammenhänge, sondern auch die Änderungsabhängigkeiten zwischen Realweltobjektmodellen in einer expliziten und damit auch für andere Instanzen sichtbaren Form. Diese Zusammenhänge zwischen originalen und ausgewerteten Daten können dabei wie bei *Henricksen* in einer eher abstrakten Form durch spezielle Assoziationstypen lediglich *repräsentiert* werden (vgl. 4.6.2 auf S. 119), oder es können innerhalb der Assoziation konkrete Codeblöcke angegeben werden, welche die betreffende Transformation direkt *durchführen*. Auf die letztgenannte Möglichkeit wird an späterer Stelle in Kap. 5.4.5.3 („Aktive Assoziationen“, S. 161) als Teil des in dieser Arbeit entwickelten Kontextmodells detaillierter eingegangen.

Vorteilhaft an *Aktiven Assoziationen* ist die Möglichkeit, den Datenpfad, d. h. die Ablaufstruktur, welche Informationen wie aus welchen anderen Informationen berechnet werden, relativ direkt abbilden zu können. Da die dabei anfallenden Zwischenschritte und Teiloperationen in einer separaten, schrittweisen, semantisch aussagefähigen Form dargestellt werden, ergibt sich eine „Karte“, auf der nicht nur sämtliche bekannten Informationen verzeichnet sind, sondern auch die zwischen diesen existierenden Zusammenhänge und Verarbeitungsschritte (vgl. Abb. 43). Dafür muß jedoch die von der Objektorientierung bekannte Kapselung von Code und Daten in Objekten aufgeweicht werden, was bei sehr großen Modellen zu Komplexitätsproblemen führen kann, denen durch geeignete modulare Konzepte anderer Art zu begegnen ist.

4.3.4 Transformatorenpfade

HONG versuchte, die Komposition von Datenpfaden durch *Aktive Assoziationen* weiter zu automatisieren, indem die einzelnen Verarbeitungsschritte nicht manuell festgelegt, sondern zur Laufzeit automatisch bestimmt werden (vgl. Abb. 44 auf der gegenüberliegenden Seite). Dabei werden von Client-

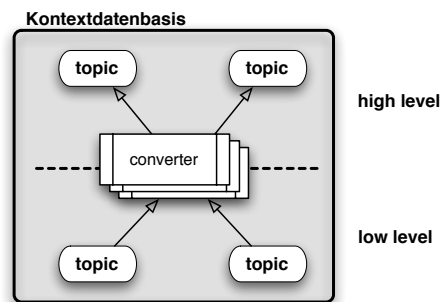


Abbildung 44: Automatische Generierung eines passenden Transformationspfades

Anwendungen angeforderte *high-level*-Ansichten automatisch über beliebig miteinander kombinierbare Transformatoren aus dynamisch bestimmten *low level*-Ausgangsdaten bestimmt.

Die dadurch ermöglichte *deklarative* Kontextabfrageschnittstelle für Clientanwendungen erlaubt die enorm flexible und generische Verwendung von Kontextinformationen in verschiedenen, auch unvorhergesehenen Anwendungsfällen analog zu den Möglichkeiten von SQL in der klassischen Datenhaltung. Allerdings ist dies nur in abgegrenzten, vollständig überblickbaren Kontextszenarien mit einer komplett und widerspruchsfrei formalisierbaren Komplexität möglich. Darüber hinaus ist nicht sichergestellt, dass mit dieser Methode auch wirklich die gewünschten Ergebnisse berechnet werden, da hierbei eine Auswahl aus z. T. widersprüchlichen Kontextinformationen erforderlich werden kann. In komplexen kontextsensitiven Szenarien kann eine solche vollautomatisierte Kontextinformationstransformation daher höchstens eine Teilrolle spielen. HONG musste den Anwendungsbereich seiner Arbeit daher auf die relativ begrenzte und hinsichtlich dieser Arbeit uninteressante Verarbeitung von Identitäts- und Lokalisierungsinformationen einschränken (vgl. [Hon⁺03, S. 34]). Neben den heute noch existierenden technischen Grenzen hängt die Entscheidung, ob die Verarbeitung von Kontextinformationen überhaupt derart automatisiert erfolgen *soll* auch von den Rahmenbedingungen im entsprechenden Anwendungsfall ab. Schließlich ist eine vollautomatisierte Auswahl und Verarbeitung von Kontextinformationen prinzipiell unkontrollierbar, was zu möglicherweise falschen Schlußfolgerungen des Systems aufgrund fehlenden Domänenwissens etwa hinsichtlich zu berücksichtigender Randbedingungen führen kann.

4.4 Historisierung von Informationen

Wie bereits in Anforderung 3c („Darstellung der Historie von Kontextvariablen“) dargestellt wurde, stellt die Historie von Kontextinformationen eine zusätzliche Kontextdimension dar, deren Analyse Trends zur Prognose zukünftiger Ereignisse offenbart. Die Darstellung der Historie von Kontextinformationen stellt daher eine elementare Anforderung an ein Kontextmodell dar und wird von allen Kontextmodellen unterstützt, wobei abhängig von den Eigenschaften der betreffenden Anwendungsdomäne zwei Ansätze unterschieden werden können.

Zumeist wird die Historie der Kontextinformationen nicht im eigentlichen Kontextmodell vorgehalten, sondern unterhalb dessen innerhalb der jeweiligen Zugriffsschnittstellen, d. h. auf der Zugriffsebene (vgl. Abb. 45 auf der nächsten Seite). So würde die Zugriffsschnittstelle Z_1 , welche den Zugriff auf die Informationen in der Kontextquelle Q_1 , die im Kontextmodell durch das Modellelement M_1 darge-

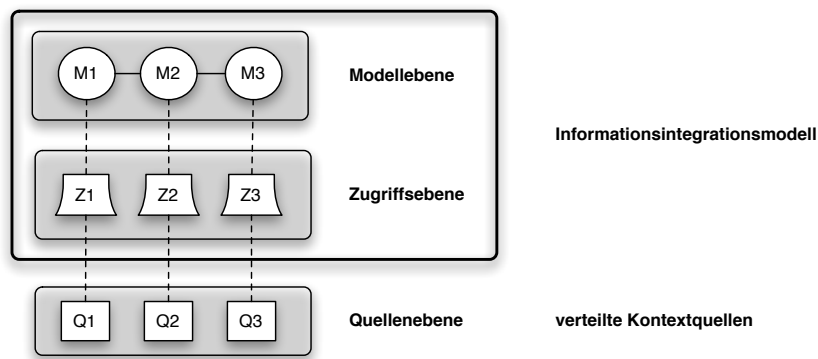


Abbildung 45: Ebenenmodell zur Integration verteilter Kontextinformationen

stellt werden, auf Anfrage den aktuellen oder aber einen historischen Wert liefern. Dadurch wird das Kontextmodell auf der Modellebene nicht durch zusätzliche historische Informationen verkompliziert. Außerdem können die einzelnen Kontextvariablen mit einer individuellen Granularität historisiert sowie auch nur bestimmte Teile des Datenmodells in beliebige vergangene Zeitpunkte zurückversetzt werden (vgl. Abb. 45, [Sch⁺99a, Dey00b]).

Die historischen Werte von Kontextvariablen können bei historieintensiven Verwendungsbereichen jedoch auch gleich innerhalb der Modellebene, d. h. direkt im eigentlichen Kontextmodell als separate Kontextdimension vorgehalten werden. HARVEL verwendet dazu sogenannte *Context Cubes*, welche auf den aus dem *Data Warehousing* (DW) bekannten *Data Cubes* basierende Kontextmodelle darstellen (vgl. Kap. 4.6.1 („Context Cubes“, S. 118)). Dabei kann auf die aus dem Data Warehousing und Online Analytical Processing (OLAP) bekannten Verdichtungs- und Analysefähigkeiten zurückgegriffen werden, wobei jedoch auch die gleichen prinzipiellen Performanz- und Kapazitätsproblemen auftreten, was diesen Ansatz nur für abgegrenzte Anwendungsfälle und nur schwer in mobilen bzw. pervasiven Domänen einsetzbar macht.

HENRICKSEN adressiert diese Schwachstelle durch einen auf einem semantischen Netz basierenden Ansatz, in dem neben inhaltlichen Assoziationsarten auch temporale Assoziationen dargestellt werden können, die auf vergangene Attributwerte verweisen. Während dadurch die zu speichernde Datenmenge signifikant reduziert werden kann, da nur noch relevante historische Kontextinformationen dargestellt werden, leidet dadurch die Übersichtlichkeit des Modells drastisch, da an dieser Stelle verschiedene, eigentlich orthogonale Konzepte, nämlich der Aufbau und die Historie eines Modells miteinander vermengt werden (vgl. Kap. 4.6.2 („Henricksen“, S. 119)).

4.5 Informationsmetamodelle

Bevor auf spezialisierte Kontextintegrationsmodelle eingegangen wird, werden an dieser Stelle zunächst klassische Informationsmetamodelle hinsichtlich ihrer Tauglichkeit zur Darstellung integrierter Kontextinformationen analysiert. Diese stellen selbst keine konkreten Datenmodelle dar, sondern einen Rahmen zur Definition eigener Datenmodelle. Um eine größtmögliche Flexibilität zu erreichen, werden bei Metamodellen sehr viele Funktionen erst von den konkreten Modellen spezifiziert. Bei deren Bewertung wird daher von konkreten Implementationsdetails abstrahiert, d. h. nicht auf mittels dieser Metamodelle kon-

kret implementierte Kontextmodellierungstechniken eingegangen, sondern auf die generelle Eignung der jeweiligen Technologie hinsichtlich der zu unterstützenden Funktionalitäten. Zur Bewertung wird das in Kap. 2.4.1 („Bewertungsschema“, S. 25) vorgestellte Bewertungsschema verwendet. Die detaillierten Ergebnisse der Bewertung der einzelnen Technologien finden sich in Anhang A („Detaillierte Bewertungen verwandter Arbeiten“, S. XXXVII), während an dieser Stelle nur deren Ergebnisse dargestellt werden.

4.5.1 Relationales Datenmodell

In vielen, insbesondere kleineren kontextbezogenen Anwendungen werden die Kontextinformationen zusammen mit den anderen Anwendungsdaten in einem normalen Datenbanksystem gespeichert und daher im *relationalen Datenmodell* dargestellt (vgl. u. a. [MaJo03], [FaCl04, S. 3], [Hen⁺02, S. 171], [Codd70]).

Relationale Datenbanken sind heutzutage in vielen Formen verfügbar und weithin bekannt. Sie stellen mit SQL eine standardisierte Schnittstelle zur Außenwelt zur Verfügung, wodurch deren Inhalte problemlos interpretiert und in andere Systeme integriert werden können. Fortgeschrittene Konzepte wie Trigger, welche eine beim Eintreten bestimmter Bedingungen automatisch vom Datenbankserver auszuführende SQL-Anweisung darstellen, ermöglichen die zeitnahe Einbindung der Inhalte relationaler Datenbanken in externe Anwendungen. Auch performanceseitig sind relationale Datenbankmanagementsysteme bereits sehr weit optimiert und dadurch in der Lage, auch große Mengen von Ereignissen in schneller Folge zu verarbeiten.

HENRICKSEN stellte im Rahmen der Analyse der Tauglichkeit des bei der relationalen Datenhaltung verwendeten Entity-Relationship-Modells (ERM) hinsichtlich der Modellierung kontextbezogener Informationen fest, dass das für die strukturierte und redundanzfreie Darstellung und die Verarbeitung großer Mengen gleichartiger Daten von einwandfreier Datenqualität ausgelegte ERM nicht der natürlicherweise einsichtigen Vorgehensweise bei der Modellierung und Verarbeitung von kontextbezogenen Informationen entspricht, da viele der in Kap. 2.4 („Anforderungsanalyse“, S. 25) dargestellten Anforderungen durch das relationale Datenmodell nur umständlich oder nur durch proprietäre Erweiterungen des Standards dargestellt werden können, u. a. aus dem Bereich der Modellierung des dynamischen Charakters von kontextbezogenen Informationsstrukturen.

Die in Anhang A.1 auf S. XXXVII detaillierter dargestellte Bewertung des ER-Modells offenbart, dass sich dieser Ansatz eher zur allgemeinen Darstellung strukturierter Daten eignet und große Defizite bei den kontextbezogenen Aspekten sowie der Integration von Informationen aufweist. Konform mit HENRICKSEN kann an dieser Stelle daher festgestellt werden, dass relationale Datenbanken eher zur Speicherung von einfachen Kontextdaten geeignet sind, anspruchsvollere Projekte jedoch auf einer eigenen Methodik, welche die Eigenschaften von Kontextdaten gezielt adressieren, basieren sollten (vgl. [Hen⁺02]). Diese Ansicht wird u. a. auch von HARVEL geteilt, der vor allem auf die fehlenden Möglichkeiten zur Darstellung von Interpretationenszusammenhängen innerhalb einer relationalen Datenbasis verweist (vgl. [Har⁺03, S. 1]).

4.5.2 Objektorientiertes Datenmodell

Moderner als das Entity-Relationship-Modell sind objektorientierte Datenbankparadigmen. Zu speichernde Informationen werden dabei in einer Objektmodellierungstechnik wie der *Unified Modeling Language* (UML) dargestellt, die Umsetzung erfolgt dann in einer konkreten objektorientierten Programmiersprache wie z. B. *Java* (vgl. [Atk⁺95]). Dies ermöglicht die Integration von Datenhaltungskonzepten in den Anwendungscode, was meist in Form von persistenten Objektstrukturen realisiert wird.

HENRICKSEN hat die Eignung objektorientierter Konzepte zur Modellierung von Kontextdaten eingehender untersucht und ist zu ähnlichen Schlußfolgerungen wie beim relationalen Datenmodell gekommen (vgl. [Hen⁺02]). An dieser Stelle sollen in dieser Arbeit jedoch genauere Untersuchungen durchgeführt werden, die in Anhang A.2 auf S. XL detailliert dokumentiert sind und an dieser Stelle zusammengefasst werden. Objektorientierte Datenhaltungskonzepte können besser mit einigen der beim relationalen Datenmodell besprochenen Problemfelder umgehen. Dazu gehören neben einer besseren allgemeinen Verwendbarkeit die gute Unterstützung dynamischer und multidimensionaler Informationsstrukturen.

Allerdings ist das objektorientierte Datenmodell nicht auf die spezifischen Eigenschaften von Kontextinformationen spezialisiert. Objektorientierung stellt zwar ein formales Metamodell zur Modellierung von Klassen zur Verfügung, aber keines zur Modellierung von Kontextklassen mittels Objektklassen (vgl. [Hen⁺02, S. 171]). Offen bleibt daher auch die Unterstützung von Daten mit unterschiedlicher Qualität und Sicherheitsanforderungen sowie Prinzipien der Gestaltung endnutzertauglicher Nutzerschnittstellen für objektorientierte Datenbanken. Durchwachsen sind ebenfalls die Fähigkeiten zur Integration heterogener Kontextinformationen. Auch fällt die maschinelle Auswertung objektorientierter Datenstrukturen etwa durch Expertensysteme schwer.

Objektorientierung stellt ein eher *implementationstechnisches*, d. h. hinsichtlich der effizienten Implementation einer bereits definierten Problemstellung orientiertes als ein *informationsmanagementbezogenes*, d. h. hinsichtlich der Unterstützung von Informationsprozessen im wirtschaftsinformatischen Sinn orientiertes Paradigma dar. Störend macht sich der zu generische, nicht kontextbezogen spezialisierte Charakter der Objektorientierung bemerkbar. Daher ist es nicht direkt auf das in dieser Arbeit behandelte Problem des Managements von Kontextinformationen mittels eines flexiblen, semantisch aussagekräftigen Kontextmodells anwendbar. Dies schließt jedoch nicht aus, die Kontextdatenbank bzw. das Datenmodell intern objektorientiert zu *programmieren*, d. h. die Objektorientierung als Basistechnologie eines eigenen Kontextmodells zu verwenden.

4.5.3 XML

Die *Extensible Markup Language* (XML) stellt ein Metamodell zur menschen- und maschinenlesbaren Modellierung strukturierter Informationen dar. Konkrete Modelle werden i. d. R. basierend auf einer expliziten, DTD, XML-Schema oder Relax NG genannten Ontologie formuliert (vgl. [FaWa04, CIa01]). Aufgrund seiner Ausdrucksmächtigkeit und Flexibilität wird XML derzeit als universaler Datenaustauschstandard in modernen betrieblichen Anwendungssystemen, vor allem zur Interprozesskommunikation mittels *Web Services* eingeführt. Der Einsatz von XML zur Modellierung von kontextbezogenen

Informationen wird derzeit in verschiedenen Forschungsprojekten evaluiert, auf die im späteren Verlauf noch separat eingegangen wird.

XML stellt genau wie die anderen beiden Metamodellierungstechnologien ein sehr flexibles Mittel dar, um Daten aller Art zu strukturieren. Einer der Vorteile von XML ist neben der durch das Metamodell bereitgestellten Flexibilität die breite Palette von verfügbaren Werkzeugen zur Verarbeitung von XML. Daher können mittels XML vom technischen Standpunkt her allgemein sehr gut handhabbare Informationsmodelle entworfen werden. Auch Multidimensionalität kann hinreichend angemessen dargestellt werden

Schwächen hat XML bei der Darstellung und Unterstützung dynamischer Daten und Datenstrukturen. Durch die prinzipbedingt fehlende Möglichkeit zur Einbettung von Code ist XML in seiner standardisierten Form nicht zur Darstellung von Kontextwissen geeignet. XML erlaubt zwar durchaus flexible Strukturierungen von Daten, jedoch fehlen viele der zur Integration von heterogenen Kontextinformationen benötigten Konzepte. Ein auf XML basierendes Darstellungsformat für integrierte Kontextinformationen müsste daher fundamentale, den Vorteil der breiten Verwendbarkeit des XML-Standards zerstörende Erweiterungen des XML-Standards vornehmen.

4.5.4 RDF

Das beim *Semantic Web* verwendete *Resource Description Framework* (RDF) ist eine Methode zur Beschreibung von beliebigen Ressourcen, als welche auch Realweltobjekte aufgefasst werden können. Dabei werden Aussagen über Ressourcen in einer ontologisch definierten, maschinenlesbaren Form formuliert, womit RDF durchaus als Kontextmodell aufgefasst werden kann. Allerdings ist es nicht direkt zur Modellierung von Realweltobjekten, sondern von beliebigen Dingen konzipiert und stellt daher eher allgemeine und keine kontextspezifischen Strukturierungsmittel zur Verfügung.

Ein auf RDF basierendes Kontextmodell stellt neben den Realweltobjekten auch die einzelnen Kontextvariablen als Ressourcen dar, wobei die Zuordnung von Kontextvariablen zu den entsprechenden Realweltobjektmodellen sowie die Verbindung verschiedener Realweltobjektmodelle untereinander über separate Aussagen, sogenannte *statements* realisiert wird. Ein auf RDF basierendes, zur Darstellung von räumlichen und technischen Kontextinformationen verwendetes Kontextmodell wird im von RAKOTONIRAINY, LOKE und ZASLAVSKY entwickelten *m3*-Framework verwendet vgl. [Rak⁺01, S. 15 f]).

Durch RDF lassen sich allgemein sehr gut handhabbare Kontextmodelle erschaffen (vgl. Anhang A.4 auf S. XLIV). Auch Aspekte der Multidimensionalität, Datenqualität sowie Zugriffsrechte auf Kontextinformationen ließen sich mit RDF-Konstrukten einfach realisieren. Konzeptionelle Lücken weist RDF jedoch bei dynamischen Elementen, der Integration von externen Informationen sowie der Darstellung von Transformationen der Inhalte der Datenbasis auf. Auf RDF basierende Modelle ließen sich mit entsprechenden Erweiterungen durchaus endnutzertauglich gestalten, jedoch gibt es in diesem Bereich geeignetere Ansätze. Negativ sticht an dieser Stelle die unintuitive Konstruktion eines Weltmodells ausgehend von den Kontextvariablen anstatt von den Realweltobjekten hervor.

Insgesamt lässt sich feststellen, dass viele Elemente eines Kontextmodells in einer auf RDF basierenden Lösung dargestellt werden können, wobei die notwendigen Erweiterungen mitunter jedoch recht

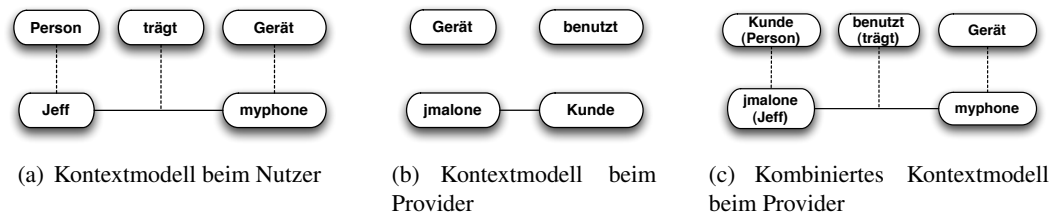


Abbildung 46: Merging von Kontextontologien durch POWER

umständlich und kompliziert ausfallen würden und durchaus fundamentale Veränderungen des Standards erfordern würden.

4.5.5 Topic Maps

Topic Maps (TM) sind ein ISO-Standard zur Darstellung von Wissen über die semantische Bedeutung sowie die Zusammenhänge von heterogenen Informationsmengen (vgl. [Schw04, Bie⁺99]). Genau wie durch RDF können damit verteilte und heterogene Webdokumentteile und andere Informationen strukturell in größere Zusammenhänge, d. h. deren Kontext, eingeordnet werden.

Topic Maps sind ein integraler Bestandteil des *Semantic Web* (vgl. 4.2.4 („Semantic Web“, S. 106)), bei welchem Wissen über Ressourcen in einer externen, d. h. von den beschriebenen Ressourcen getrennten Schicht, der sogenannten *Kontextschicht* dargestellt wird (vgl. Abb. 39 auf Seite 107). Eine Topic Map stellt also im Prinzip ein Organisationsmodell für ansonsten beliebig unstrukturierte, jedoch adressierbare Ansammlungen unterschiedlicher Informationsfragmente dar, welches auf einem Stellvertreteransatz basiert, wobei sogar die Beziehungen zwischen den Realweltobjektstellvertretern modelliert werden können (vgl. [Hut⁺01]). Damit können sie prinzipiell auch zur maschinenlesbaren Beschreibung und Integration von Informationen über Dinge der realen Welt, d. h. ihren Eigenschaften und Beziehungen untereinander, also zur Darstellung von Kontextdaten verwendet werden (vgl. [Powe03], [HaSc04, S. 111]). Dabei modellieren *Topics* die zu beschreibenden Realweltobjekte, *Assoziationen* deren Zusammenhänge, während *Occurrences* auf die als Attribute aufgefassten externen Informationen verweisen.

Diese Idee, Topic Maps als Ontologie zur Organisation von verteilten Kontextdaten zu verwenden, geht auf POWER zurück (vgl. [Powe03]). Dabei wird ausgehend von der Überlegung, dass es aufgrund der Komplexität und fehlenden Koordination aller Beteiligten kein *weltweites* Schema für Kontextdaten geben kann vorgeschlagen, Kontextmodelle auf *Organisationsebene*, d. h. innerhalb einzelner Unternehmen zu modellieren und diese heterogenen Kontextdatenstrukturen dann bei Bedarf durch eine Metaebene in Form von Topic Maps zu verbinden, welche die passenden Informationen in den verschiedenen Datenmodellen in einen Zusammenhang bringt. Durch diesen Ansatz wird es möglich, die Datenbestände verschiedener Organisationen aus einer gemeinsamen, anwendungs- und subjektübergreifenden Sichtweise zu betrachten, wobei die diversen Ontologien miteinander vermengt werden. Als Beispiel wird in Abb. 46(a) das von einem Endkunden bereitgestellte Kontextmodell über seine persönliche Situation mit dem in Abb. 46(b) dargestellten Kontextmodell auf Seiten eines Mobilfunkproviders zu der in Abb. 46(c) dargestellten integrierten Ansicht vereinigt (vgl. [Powe03, S. 10]).

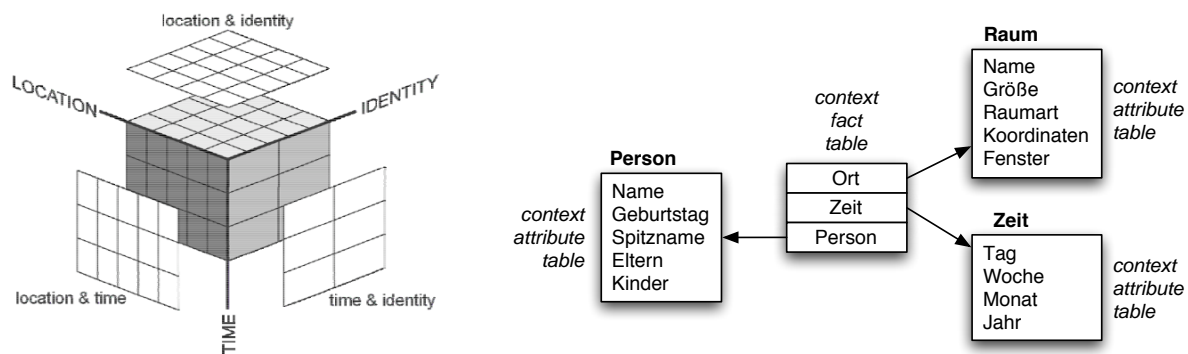
Topic Maps ermöglichen allgemein bis auf eine Ausnahme gut handhabbare Kontextmodelle, die auch angemessen mit der Multidimensionalität von Kontextstrukturen umgehen können und stellen ein zumindest teilweise endnutzertaugliches Strukturierungsverfahren für Informationen dar. Hervorzuheben ist der für die gegebene Problemstellung geeignete prinzipielle Ansatz von TM, der viele kontextbezogenen Fragestellungen trotz einer nicht kontextspezifischen Konzeption des Standards in einer für die Kontextdarstellung geeigneteren Weise adressiert als dies durch andere Ansätze erfolgt. So werden mit *Scopes* und der Möglichkeit, mehrere Topic Maps miteinander zu vereinigen (*Merging*) sehr geeignete modulare Konzepte zur Verfügung gestellt. Ebenfalls relativ einzigartig ist die einfache, an der hypertextartigen Struktur des *World Wide Web* orientierte Struktur von TM, die sich an der Lebenswelt der Endnutzer, die heute durchaus Internet und Browser, aber noch keine Datenbankkonzepte beinhaltet, orientiert.

Durchaus gravierende Probleme in fast allen Anforderungsbereichen ergeben sich jedoch durch die bereits bei RDF kritisierte Unfähigkeit zur Darstellung methodischer Elemente sowie die nur bei Topic Maps fehlende Fähigkeit zur Darstellung von Inhalten referenzierter Informationen innerhalb des Modells, denn Topic Maps können diese nur verlinken und stellen damit lediglich eine *strukturell*, aber nicht *inhaltlich* integrierte Sichtweise auf die heterogenen Informationen dar (vgl. [Fehr04, S. 10]). Eine gezielt hinsichtlich dieser Schwachstellen orientierte Erweiterung des Topic-Map-Standards könnte daher zu einem hinsichtlich sämtlicher geforderten Belange sehr geeigneten Kontextmetamodell führen.

4.5.6 Ontologien

Um die Informationen aus unterschiedlichen Quellen, welche alle die selbe Domäne beschreiben, in einen Zusammenhang bringen zu können, ist es notwendig, die semantische Bedeutung aller dabei verwendeten Begrifflichkeiten, d. h. ein repräsentatives Vokabular zur Beschreibung der betreffenden Realweltdomäne explizit zu modellieren. Nur dadurch kann sichergestellt werden, dass sämtliche beteiligten Instanzen sicher sein können, unter den selben Begriffen die selben Konzepte zu verstehen. Dies ist die Aufgabe von *Ontologien*, welche die *Konzeptualisierung* von Modellen beschreiben, indem sie festlegen, welche Dinge und Konzepte im Begriffsverständnis einer Instanz welchen Namen besitzen (vgl. [Grub93a, S. 1 f]). Semantisches Bedeutungswissen wird den zu beschreibenden Informationen dabei durch die Zuordnung zu einem ontologischen Begriff, welcher wiederum ein Realweltkonzept repräsentiert, verliehen (vgl. [Schw01]).

Ontologien als Definitionen von repräsentativen Vokabularen zur Beschreibung von Ausschnitten der Welt werden im Informations- und Wissensmanagement vor allem als Strukturierungsmittel und zur Beschreibung auszutauschender Informationsbestände, d. h. den zwischen zwei unabhängigen Instanzen auszutauschenden strukturierten Daten verwendet, denn durch die Verwendung einer gemeinsamen Ontologie können zwei ansonsten komplett unabhängige Instanzen Informationen über einen Diskursbereich austauschen, ohne auf eine globale Beschreibung dieses Diskursbereiches zurückgreifen zu müssen (vgl. [Schw01, Guar98]). Eine explizite ontologische Beschreibungen der Struktur und Bedeutung der Informationsbasis ist für deren kollektive Nutzung also absolut notwendig (vgl. [Grub93a, S. 2]). Die Kenntnis aller beteiligten Ontologien inklusive der notwendigen interontologischen Informationen, d. h. Informationen darüber, welche Begriffe Synonyme, Hyper- und Hyponyme darstellen, erlaubt fernerhin die automatisierte Übersetzung von strukturierten Informationsbeständen in andere Vokabulare bzw. Re-



(a) Context Cube zur Darstellung, wer sich wann wo befindet

(b) Zur Definition des in Abb. 47(a) dargestellten Context Cubes verwendetes Star-Schema

Abbildung 47: Context Cubes

präsentationen, wobei dies bei deckungsungleichen Ontologien nur mit einem gewissen Informationsverlust durchgeführt werden kann. In dieser Arbeit werden Ontologien vor allem als Strukturinformationen und Beschreibung der Semantik von Realweltobjekten verwendet.

Ein Mittel zur Darstellung von Ontologien stellen Semantische Netze dar, wobei Knoten Konzepte, d. h. Begriffe oder Themen und Kanten Beziehungen zwischen diesen Konzepten symbolisieren (vgl. [Schw01]). Dieser Ansatz wird in Kap. 5.2.2 („Universen“, S. 140) aufgegriffen.

4.6 Kontextmodelle

Nach der grundlegenden Einordnung dieser Arbeit und der Bewertung allgemein verwendbarer Datenmodelle hinsichtlich der gegebenen Aufgabenstellung werden im folgenden spezialisierte Kontextinformationsmodelle in entsprechender Weise analysiert. Dabei wird jeweils ein repräsentativer Vertreter eines Informationsmodellansatzes vorgestellt und hinsichtlich der für diese Arbeit relevanten Bewertungskriterien analysiert. Neben einer kurzen Beschreibung der Funktionalität des jeweiligen Ansatzes werden an dieser Stelle nur die grundlegenden Ergebnisse dieser Analyse vorgestellt; für detailliertere Untersuchungsergebnisse wird auf den Anhang A auf S. XXXVII verwiesen. Zur Bewertung wird das in Kap. 2.4.1 („Bewertungsschema“, S. 25) vorgestellte Bewertungsschema verwendet.

4.6.1 Context Cubes

Harvel modelliert Kontextdaten durch Data Warehouse-Technologien, insbesondere dem Data Cube-Modell. Bei dem von ihm entwickelten *Context Cube* werden die Realweltobjekte durch *context attribute tables* dargestellt. *Context fact tables* verbinden diese per Fremdschlüssel zu multidimensionalen Datenräumen, welche aus verschiedenen Kontextdimensionen aufgespannt werden (vgl. Abb. 47). Der Fokus des Modells liegt dabei eindeutig auf der sehr engen Integration weniger Kontextdimensionen und der Aufbereitung der so dargestellten Kontextinformationen für statistische Analysevorhaben (vgl. [Har⁺03]).

Hinsichtlich der Anforderungen dieser Arbeit kann festgestellt werden, dass *Context Cubes* einen multidimensionalen Datenraum zur Verfügung stellen, in welchen aus externen Datenquellen beschaffte, miteinander integrierte Kontextinformationen und -ereignismeldungen eingeordnet werden können. Die Historie stellt dabei eine explizite Dimension des Datenmodells dar und wird daher modellseitig in vollem Umfang unterstützt.

Weniger geeignet ist der *Context Cube*-Ansatz zur Integration einer großen *Vielfalt* von verschiedenartigen Informationen, da sich die entstehenden multidimensionalen Datenstrukturen exponentiell vergrößern und daher schnell sehr umfangreich werden, was u. a. auch die Portabilität dieser Lösung eingeschränkt (vgl. [Har⁺03, S. 17]). Weiterhin kann die Datenbasis zwar auf verschiedenen Vergrößerungsstufen dargestellt werden, was sich jedoch nur auf die Granularität der Daten und nicht auf semantisch verdichtete Ansichten bezieht. Dynamische Elemente fehlen im Datenmodell ebenso wie Qualitäts- und Zugriffsschutzattribute. Auch bei der Darstellung der Verarbeitungskette von Kontextinformationen bestehen modellseitige Lücken. Das Modell kann nur als eingeschränkt endnutzertauglich angesehen werden.

Obwohl das CCDM viele der geforderten Anforderungen erfüllt, stellt es für die gegebene Aufgabenstellung nicht die erste Wahl dar. *Context Cubes* stellen eher ein analyseorientiertes denn ein operatives Datenformat dar, da das Datenmodell eher für umfangreiche statistische Analysen wie OLAP und Data Mining als für die effiziente Darstellung operativer, d. h. von Clientanwendungen zur Laufzeit abonnierbarer Informationen optimiert ist. Die zur Erfüllung aller Anforderungen notwendigen Änderungen am Modell wären gravierender und fundamentaler Natur.

4.6.2 Henricksen

Die Gruppe um HENRICKSEN hat ein verteiltes Infrastrukturkonzept zur Unterstützung pervasiver Anwendungen entwickelt, das neben der Übernahme typischer Laufzeitfunktionen ein Kontextinformationsmodell enthält (vgl. [Hen⁺02]).

Dieses basiert auf dem Stellvertreteransatz, d. h. Realweltobjekte werden durch Topics repräsentiert, die wiederum Attribute besitzen. Die Topics werden in semantischen Netzen organisiert, d. h. durch eine Reihe von unidirektionalen Assoziationsarten zwischen den Entities miteinander verbunden und beschrieben (vgl. Abb. 48 auf der nächsten Seite).

Dabei werden die in Tab. 13 auf der nächsten Seite dargestellten Assoziationsaspekte unterschieden, welche die Entities zu denen sie führen hinsichtlich ihrer Herkunft *beschreiben*, d. h. angeben, ob sie von Sensoren gemessen, vom Nutzer eingegeben oder aus anderen Daten abgeleitet wurden, *quantifizieren*, d. h. angeben, ob die betreffenden Assoziationen einzeln, als Menge oder alternativ auftreten sowie bestimmten Zeitabschnitten *zuordnen*, d. h. die Gültigkeit der betreffenden Assoziationen auf bestimmte Zeitabschnitte einschränkt.

Das Kontextmodell ist prinzipiell in der Lage, verteilte Kontextinformationen in einer integrierten Sichtweise darzustellen, unterstützt dies jedoch nicht im speziellen, etwa durch eine Verknüpfung der modellinternen Informationen zu den Originaldaten in den externen Datenquellen. Dank einer Anzahl dynamischer Assoziationsarten ist es zur Modellierung zumindest bedingt variabler Kontextdatenstrukturu-

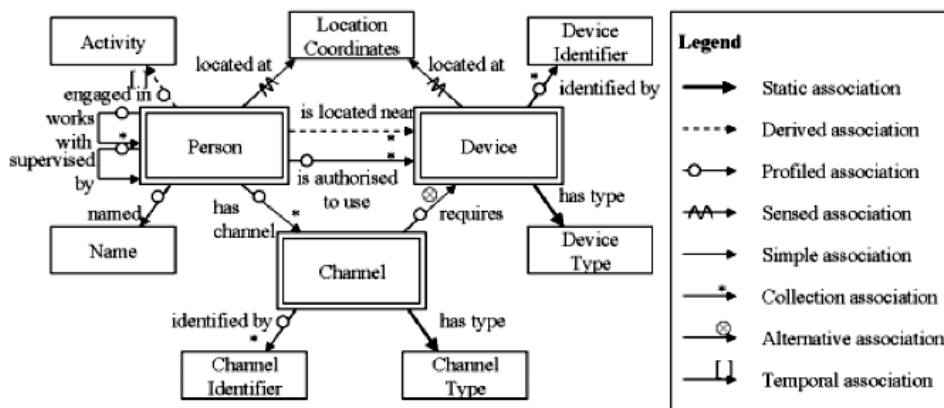


Abbildung 48: Kontextmodell nach HENRICKSEN

ren geeignet. Hervorzuheben ist in diesem Zusammenhang das ausgefeilte Qualitätsmanagementkonzept und die durch die netzartige Struktur relativ gut visualisierbaren Datenpfade.

Problematisch macht sich die fehlende Formalisierung der Modellinhalte mittels einer expliziten Ontologie bemerkbar. Aufgrund fehlender Mechanismen zur Handhabung von Komplexität droht das Modell sehr schnell sehr unübersichtlich zu werden. Dazu tragen die Vermengung der eigentlich zueinander orthogonalen Aspekte der Beschaffung, Strukturierung und Historisierung von Daten ebenso wie deren Realisierung in Form zahlreicher miteinander zu kombinierender Assoziationsaspekte sowie die fehlende Modularisierungsfähigkeit bei, was diese Technologie nur sehr bedingt für Endanwender tauglich macht. Als Beispiel sei Abb. 48 gegeben, deren Darstellung bereits bei 3 Entities inklusive einiger Attribute ziemlich kompliziert und unübersichtlich ist. Problematisch ist weiterhin der Ansatz, die Deskriptoren

	Art	Beschreibung
Dynamik	<i>static</i>	normale, sicher bekannte und unveränderliche Beziehung zwischen zwei Entities
	<i>sensed</i>	beschreibt, dass die Information des Zielentity von einem Sensor erfasst wird und daher ungenau und veraltet sein kann
	<i>derived</i>	beschreibt, dass das Zielentity aus einem anderen Entity abgeleitet wurde und daher dessen Datenqualität erbt
	<i>dependency</i>	beschreibt, dass eine Assoziation auf externem Weg von einer anderen Assoziation abhängt, d. h. bei Änderung dieser Assoziation auch neu bestimmt werden muß
Kardinalität	<i>profiled</i>	beschreibt, dass die Information vom Nutzer angegeben wurde und im Prinzip sicher bekannt ist, aber aufgrund fehlender Aktualisierung veraltet sein kann
	<i>simple</i>	jedes Entity kann nur einmalig in der Assoziation auftreten
	<i>collection</i>	es kann mehrere Versionen vom Zielentity geben, wobei <i>alle</i> Versionen verwendet werden können
	<i>alternative</i>	es kann mehrere Versionen vom Zielentity geben, wobei jeweils nur <i>eine</i> Version verwendet werden kann
	<i>temporal</i>	die alternativen Zielentities sind zu unterschiedlichen Zeiten aktiv

Tabelle 13: Assoziationsaspekte im Model von HENRICKSEN

Element	Beschreibung
<i>Context Widget</i>	Standardisierte Zugriffsschnittstelle auf einen Sensor bzw. Aktor. Stellt eine einzelne Kontextvariable inklusive deren Historie bereit und sendet von anderen Anwendungen abonmierbare Änderungsnachrichten (vgl. [Dey00b, S. 39 ff]).
<i>Context Interpreter</i>	transformieren low-level Kontextinformationen zu höherwertigeren Kontextinformationen (vgl. [Dey00b, S. 40 ff])
<i>Context Aggregator</i>	integrieren von verschiedenen Widgets kommende Kontextdaten zu Entities, Zugriffsschnittstelle für Client-Anwendungen (vgl. [Dey00b, S. 41]).
<i>Discoverer</i>	findet passende Widgets bzw. Aggregatoren (vgl. [Dey00b, S. 57 ff])

Tabelle 14: Elemente des Kontextmodells des *Context Toolkit*

von Topics nicht bei den jeweiligen Topics, sondern in den Assoziationen, die zu diesen Topics führen, zu speichern, da die Entities dadurch selbst nicht beschrieben werden und deren Beschreibung somit nicht in anderen Zusammenhängen sichtbar ist. Auch wird keine Rücksicht auf die Multidimensionalität der zu modellierenden Domäne genommen, noch werden Zugriffsbeschränkungen im Modell berücksichtigt. Auch enthält das Modell nicht sämtliche benötigten Informationen. So ist es zwar möglich, eine Reihe von alternativen Assoziationen anzugeben, jedoch können keine weiterführenden Entscheidungskriterien angegeben werden, etwa wann welche Assoziation auszuwählen ist.

Das Modell von *Henricksen* stellt außer einigen typisierten Assoziationsarten nicht wirklich neue Konzepte vor, die bei der Modellierung kontextbezogener Informationen helfen und leidet unter einer Reihe bereits dargestellter prinzipieller Schwächen. Die Erweiterung dieses Modells hinsichtlich der in Kap. 2.4 („Anforderungsanalyse“, S. 25) geforderten Fähigkeiten würde die bereits jetzt unübersichtliche Assoziationsvielfalt noch einmal drastisch erhöhen, so dass dieses Modell zwar als konzeptuelle, jedoch nicht als konkrete Basis für das in dieser Arbeit entwickelte Darstellungsformat fungieren sollte.

4.6.3 Context Toolkit

Das von ABOWD und DEY entwickelte *Context Toolkit* stellt das erste *generische* Framework zur Unterstützung kontextsensitiver Anwendungen dar. Es unterstützt die Erfassung von Kontextdaten mittels eines abstrakten Sensor- und Aktorenmodells sowie die Verarbeitung und Agglomeration der erfassten Kontextinformationen zu *high-level*-Kontext. Dazu bietet es eine Reihe von Abstraktionsmechanismen, die in Tabelle 14 dargestellt werden. Das Context Toolkit verfolgt einen *bottom-up*-Ansatz bei der Definition des Verarbeitungspfades von Sensoren zu high-level-Kontextinformationen, d. h. die Inhalte einzelner, in Form von *Widgets* dargestellter Kontextvariablen werden durch Nutzung von *Interpretern* zunächst in höherwertigere Sichtweisen transformiert und erst dann werden die bekannten Informationen mittels *Aggregatoren* zu Realweltobjektrepräsentationen zusammengefasst (vgl. Abb. 49 auf der nächsten Seite, [Dey⁺99a], [Dey00b]).

Das Kontextinformationsmetamodell des *Context Toolkit* ist bedingt gut verwendbar, wobei jedoch wichtige Funktionen, etwa Beziehungsinformationen zwischen Entities nicht dargestellt werden können, so dass dieses Kontextmetamodell eher eine Zugriffsschnittstelle oder Verdichtungsebene für sensorisch erfasste Kontextdaten denn eine Darstellungs- und Integrationsplattform für kontextbezogene Informa-

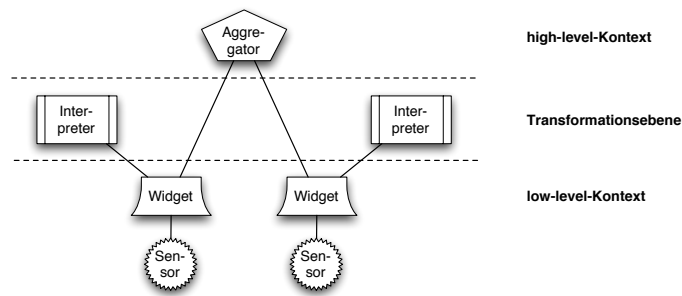


Abbildung 49: Schichtenarchitektur des Kontextmodells des *Context Toolkit*

tionen darstellt. Es ist für die Verarbeitung dynamischer Informationen konzipiert, lässt jedoch Mechanismen zur Adaption an veränderte Umstände vermissen. Hinsichtlich der Problemstellung der Integration heterogener kontextbezogener Informationen werden einige, jedoch keine neuartigen Mechanismen bereitgestellt, wodurch mit den gegebenen Mitteln keine wirklich brauchbaren Ergebnisse in diesem Bereich erzielt werden können. Das Modell kann für endnutzerprogrammierergerechte Nutzerschnittstellen verwendet werden.

Nachteilig in vieler Hinsicht macht sich das Fehlen von Assoziation bemerkbar. Auch fehlen Mechanismen zur Modellierung der Datenqualität bzw. -sicherheit. Aufgrund des frühen Entwicklungszeitpunktes des Modells sind viele später als relevant erkannte Funktionen nicht im Modell vorgesehen, etwa explizite Ontologien oder Mechanismen zur Integration externer, heterogener Kontextinformationen.

Zusammenfassend kann festgestellt werden, dass das Kontextinformationsmetamodell des *Context Toolkit* vor allem hinsichtlich der komfortablen Bereitstellung von Sensormesswerten konzipiert ist, viele der zur Integration heterogener Kontextinformationen notwendigen Konzepte nicht enthält und daher nur mit gravierenden Erweiterungen für dieses Problemgebiet tauglich gemacht werden könnte.

4.6.4 Context Fabric

HONG stellt mit *Context Fabric* ein Kontexterfassungssystem mit einer *deklarativen* Abfrageschnittstelle analog zu SQL in der konventionellen Datenhaltung bereit, wodurch Kontextinformationen möglichst einfach in Anwendungsprogramme integriert werden sollen (vgl. [Hong01, HoLa01, Hong02]). Dazu formulieren Clientanwendungen durch eine Anfrage über die Kontextabfrageschnittstelle von *Context Fabric* ihren Bedarf nach *high-level*-Kontextinformationen, worauf das System automatisch die zur Beantwortung der Anfrage geeignetesten Sensoren sowie Transformationsschritte von *low-level* zu *high-level*-Kontextdaten bestimmt und damit quasi vollautomatisch eine Abfrage der Realwelt durchführt. Als Beispiel ist in Abb. 50 auf der gegenüberliegenden Seite die Anfrage nach in der Nähe befindlichen Kinos dargestellt, welche über einen Vergleich ihrer Postleitzahlen bestimmt werden, wobei die Postleitzahl des Standortes des Nutzers abhängig von der gegenwärtigen Situation mittels GPS, der Funkzelle seines Mobiltelefons oder über ein indoor-Lokalisierungssystem ermittelt wird (vgl. [Hong01, S. 10]).

Dazu verwaltet es mittels eines *Sensor Management Service* eine Reihe von Sensoren sowie in einer *Automatic Path Creation*-Schicht verschiedene miteinander kombinierbare *Datentransformatoren* (vgl. Abb. 51 auf der gegenüberliegenden Seite).

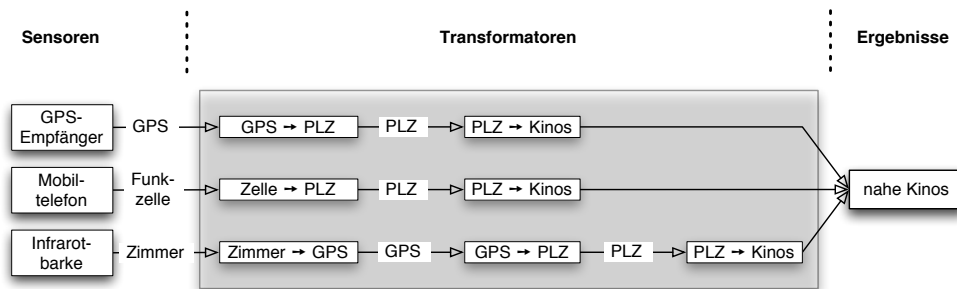


Abbildung 50: Vollautomatische Bestimmung von Pfaden am Beispiel des Findens in der Nähe befindlicher Kinos

Context Fabric schneidet in allen Anforderungsbereichen gut ab. Die besondere Stärke des Modells liegt aufgrund des erst zur Laufzeit bestimmten Datenpfades sowie der ad-hoc bestimmten Sensoren in der Unterstützung veränderlicher Situationen mittels deklarativer Abfragen. Auch eignet es sich dank der Möglichkeit zur Abfrage externer Kontextinformationen bedingt zur Integration heterogener Kontextinformationen.

Die prinzipielle Stärke dieses Modells stellt jedoch auch seine prinzipielle Schwäche dar. Eine vollautomatische Bestimmung von Kontextinformationen funktioniert aufgrund der Anforderungen der dabei verwendeten Reasoning-Algorithmen nur in sehr abgegrenzten, widerspruchsfreien Szenarien. Diese Einschätzung wird auch von HONG selbst geteilt, der *Context Fabric* letztlich nur zur Bestimmung von Orten und Identitäten von Personen einsetzt. Komplexere Szenarien erscheinen zumindest in der gegenwärtig dargestellten Form nicht funktionstüchtig (vgl. [Hong01, S. 8]). Insgesamt kann nicht glaubwürdig dargestellt werden, dass durch diesen vollautomatischen Ansatz auch tatsächlich die benötigte Kontextinformation bestimmt wird, und dass dafür auch der geeigneteste Sensor gewählt wird. Die Erfahrung im Umgang mit kontextbezogenen Anwendungen zeigt, dass diese Domäne von vielen Ausnahmen und Sonderfällen geprägt ist, die alle formalisiert modelliert werden müssten. Es steht zu bezweifeln, dass dieses notwendige Domänenwissen in vollständiger und widerspruchsfreier Form in einem ökonomisch sinnvollen Rahmen dargestellt werden kann. Weiterhin ergeben sich durch den nichtmodularen Ansatz Skalierungsprobleme.

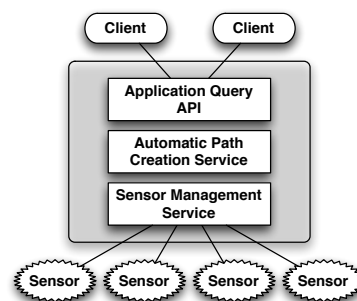
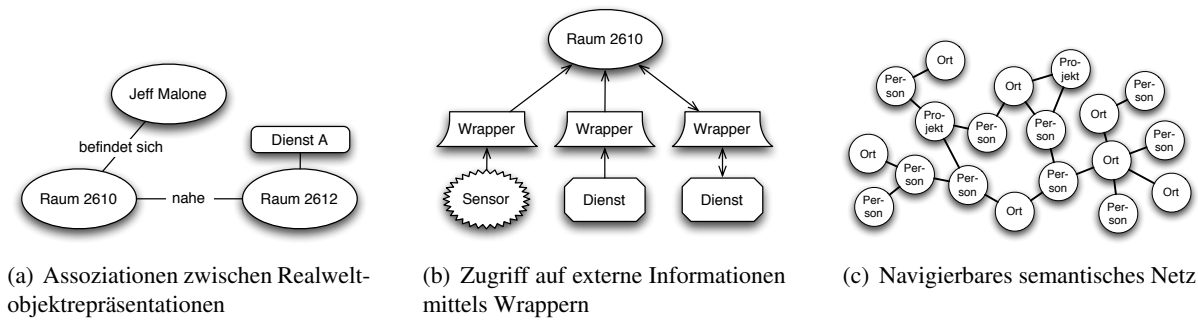


Abbildung 51: Struktur des *Context Fabric*

Abbildung 52: *Context Shadow* von JONSSON

Insgesamt stellt dieser Ansatz eine sehr generische, jedoch in der Praxis kaum realisierbare Idealvorstellung für ein Kontextabfragemodell dar, das konzeptuell für den in dieser Arbeit angenommenen Anwendungsfall auf jeden Fall zu hoch angesiedelt ist.

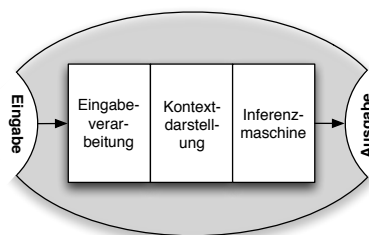
4.6.5 Context Shadow

Der Context Shadow-Ansatz verlinkt Repräsentationen von Realweltobjekten durch verschiedene Assoziationen, wodurch eine semantisch navigierbare Topologie von Realweltobjektrepräsentationen entsteht (vgl. Abb. 52(a)). Realweltobjektrepräsentationen enthalten die jeweils zu ihren Realweltobjekten zugeordnete Informationen in Form von *Attributen*. Diese können über *Wrapper* auch auf Informationen in externen Datenquellen verweisen, wobei zwischen einer reinen Repräsentation des Attributes sowie lesendem und schreibendem Zugriff auf die darunter liegenden Sensoren und Dienste unterschieden wird (vgl. Abb. 52(b)). Die Daten werden mittels eines Tupelraums dargestellt und ausgetauscht. Ein Tupelraum ist ein Kommunikations- und Datenhaltungsverfahren, in welchem Tupel genannte Datensätze mit einem festen Datenschema, aber unterschiedlichem Inhalt gespeichert werden. Typische Operationen sind das Hinzufügen von Tupeln, das Abfragen von Tupeln gemäß einer Abfrageschablone und das Abonnieren von Meldungen über das Hinzufügen bzw. Entfernen bestimmter Tupelarten (vgl. [Car⁺94, Wyc⁺98]).

Der *Context Shadow*-Ansatz enthält grundlegende Elemente zur Lösung der gestellten Problematik, löst jedoch nicht alle Anforderungsbereiche zufriedenstellend. Dazu zählen die folgenden Punkte.

Modularität. Das gesamte Modell ist durch den Tupelspace-Ansatz nicht modular, wodurch sich in der Praxis Komplexitätsprobleme ergeben, wie in Abb. 52(c) ausschnittsweise skizziert werden (vgl. 1b („Modularität“)).

Sichtbarkeit. Ebenso sind abgegrenzte Sichtweisen im Sinne von Anf. 6j („Einschränkung der Sichtbarkeit von Konzepten“) nur sehr aufwändig zu realisieren. Tupelräume stellen eher abstrakte Räume dar, welche mit großen Mengen von Daten in der Hoffnung gefüllt werden, dass sich jeder Client die gewünschten Daten selbst herausucht. In der Regel weiß der Ersteller jedoch nicht, wer die Daten später lesen wird.

Abbildung 53: Aufbau eines *Sentient Object*

Einbettung von Code. Logik kann in Form von *Context Refiners* zwar dargestellt werden, jedoch wird dies nicht wirklich in die Datenstruktur integriert. Ein Context Refiner ist ein externer Agent, welcher Daten aus der Datenbasis liest und an anderer Stelle neue Daten einfügt (vgl. Kap. 4.3.1 („Externe Agenten“, S. 108)). Dabei wird weder der Zusammenhang zu den einzelnen Daten noch der Zeitpunkt, wann ein Refiner aufgerufen wird, sichtbar.

Sicherheit. Offen bleiben auch die wichtigen datenschutztechnischen Anforderungen. Zwar kann durch Verwendung privater Tupelräume eine unbeobachtete Kommunikation durchgeführt werden, doch ist zu bezweifeln, damit die notwendige Granularität der Zugriffsrechte realisieren zu können.

Ontologien. Ferner ist unklar, wie die verschiedenen Assoziationsarten definiert werden. Explizite Ontologien werden nicht erwähnt, sondern nur sehr wenige, vordefinierte Assoziationen, die sich auf die Recherche von Personen, Orten und Projekten beziehen, was einer globalen, nicht veränderlichen Ontologie entspräche. Es darf bezweifelt werden, mit solch einfachen Strukturierungsmitteln brauchbare kontextbezogene Dienste modellieren zu können, die eine große Menge heterogener Kontextinformationen enthalten. In diesem Stadium kann *Context Shadow* nur zur Lösung stark abgegrenzter Probleme verwendet werden.

4.6.6 Sentient Computing

Das von FITZPATRICK und BIEGEL entwickelte, auf *Sentient Objects* basierende Kontextmodell verwendet einen objektorientierten Ansatz, um Kontextdaten und die dazugehörige Methodik zu modellieren. Dabei werden kontextsensitive Anwendungen aus *Sentient Objects* genannten Objekten zusammengesetzt, die jeweils einzelne Teile der kontextbezogenen Funktionalität realisieren und miteinander interagieren.

Sentient Objects als „empfindungsfähige“ Objekte unterscheiden sich von normalen Objektstrukturen durch Umweltschnittstellen, über die sie kontextbezogene Informationen aufnehmen bzw. abgeben können (vgl. Abb. 53). Ein einzelnes Sentient Object realisiert dabei eine einzelne kontextbezogene Funktionalität, etwa die Erfassung, Verarbeitung oder die Anwendung von Kontextinformationen. Je nach Art der vom betreffenden Sentient Object unterstützten und in Tab. 15 auf der nächsten Seite dargestellten Ein- und Ausgabearten können Sensoren, Aktoren bzw. Transformatoren unterschieden werden.

Art	Eingabeart	Ausgabeart
Sensor	Realweltreize	elektronische Informationen
Transformator	elektronische Informationen	elektronische Informationen in einem anderen Format
Aktor	elektronische Informationen	Realweltreize

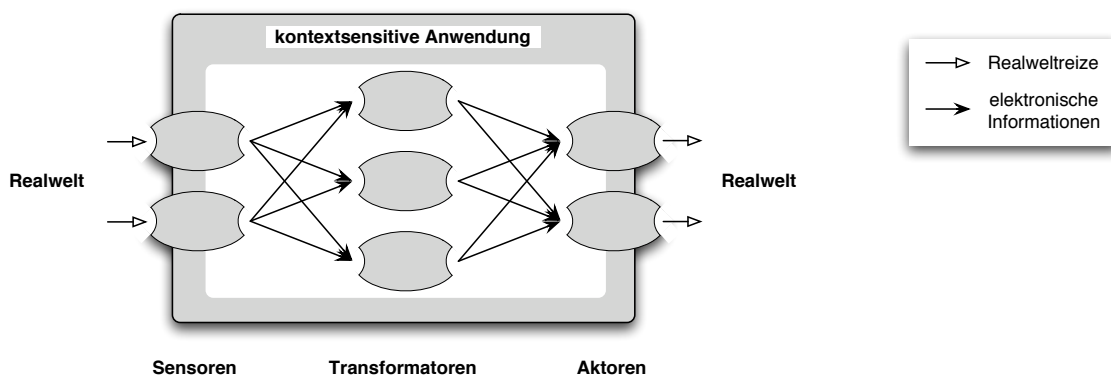
Tabelle 15: Arten von *Sentient Objects*

Kontextsensitive Anwendungen, die auf Realweltreize durch Abgabe anderer Realweltreize reagieren, werden aus mehreren, miteinander verketteten *Sentient Objects* realisiert, wobei die einzelnen Objekte miteinander durch Nachrichtenaustausch interagieren können (vgl. Abb. 54).

Innerhalb der *Sentient Objects* befindet sich neben der Repräsentation der Kontextinformationen in Form von Attributen auch der dazugehörige Auswertungscode, wobei auch Inferenzmaschinen zum Einsatz kommen können (vgl. Abb. 53 auf der vorherigen Seite, [Fit⁺02, S. 4]).

Mittels *Sentient Computing* lassen sich allgemein gut handhabbare Anwendungen erstellen, die auch gut mit multidimensionalen Aspekten umgehen können. Dynamische Kontextvariablen werden explizit unterstützt. Mittels geeigneter Nutzerschnittstellen ließen sich möglicherweise einfach zu wartende, nach dem Baukastenprinzip zusammensetzbare Anwendungen realisieren. Das und inwieweit dieses Prinzip praxistauglich umgesetzt werden kann, bleibt noch zu zeigen.

Das auf *Sentient Objects* basierende Kontextmodell stellt im Prinzip kein homogenes, sondern eine Sammlung mehrerer, in verschiedenen Objekten gespeicherter, anwendungsspezifischer Kontextmodelle dar. Zwar werden von den Autoren Überlegungen hinsichtlich der Unzuverlässigkeit von Sensormesswerten gemacht, diese finden im Kontextmodell jedoch keinen Niederschlag, was etwa in Form von qualitätsbeschreibenden Attributen geschehen könnte. Wie viele andere Kontextmodelle können mittels *Sentient Computing* heterogene Kontextinformationen zwar durchaus integriert werden, dies wird jedoch nicht im speziellen unterstützt, weshalb insbesondere die speziellen Anforderungen in diesem Bereich nicht erfüllt werden können.

Abbildung 54: Aufbau einer aus *Sentient Objects* aufgebauten kontextsensitiven Anwendung

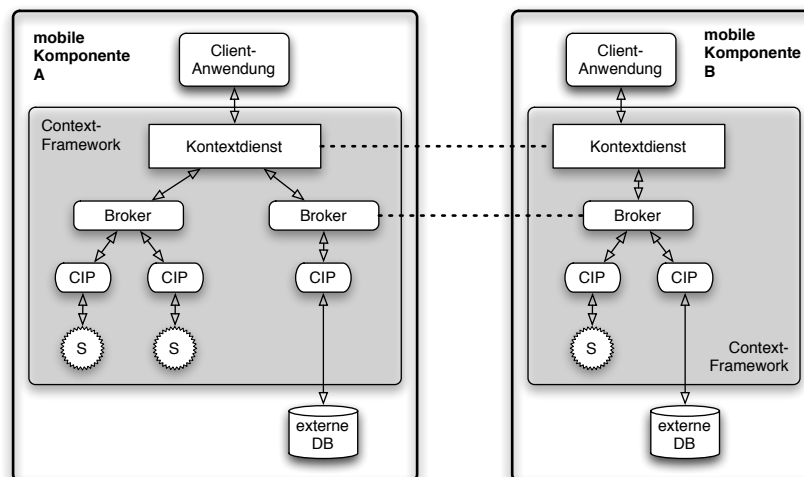


Abbildung 55: Integrationsprinzip verteilter Kontextinformationen beim *Context Framework* (vgl. [Kadn04a, S. 48])

4.6.7 Context Framework

Das von LÖSCHAU und KADNER entwickelte *Context Framework* stellt einen Infrastrukturdienst zur verteilten Kontexterfassung und -nutzung auf mobilen Endgeräten dar. Es modelliert Kontext mittels der objektorientierten Datenstrukturen einer objektorientierten Programmiersprache und verwendet entsprechend ein objektorientiertes Datenmodell. Kontextvariablen stellen dabei Attribute von Kontextobjekten dar, wobei zwischen *Wert-Kontextvariablen*, die einen Messwert enthalten und *Beziehungs-Kontextvariablen*, die Beziehungen zu anderen Kontextobjekten darstellen, unterschieden wird. Kontextinformationen werden über *Context Information Provider* (CIP) bereitgestellt, welche Softwaremodule darstellen, die entweder einen Sensor oder eine beliebige andere externe Datenquelle befragen und genau einen Kontextwert zurückliefern. Diese Informationen werden über *Broker* genannte Dienste verwaltet und transparent geräteübergreifend verfügbar gemacht, so dass sich Kontextquellen und -verbraucher auf beliebigen, auch spontan miteinander vernetzten mobilen Endgeräten befinden können (vgl. Abb. 55).

Das Kontextmodell des *Context Framework* ist auf die Verteilung von Kontextinformationen in einem peer-to-peer-Netz mobiler, spontan vernetzter Endgeräte optimiert. Entsprechend unterstützt es multidimensionale und dynamische Konzepte sowie die Darstellung von Datenqualität.

Konzepte zur Integration heterogener Kontextinformationen wurden jedoch nicht angemessen berücksichtigt. So müssen externe Kontextquellen eher umständlich über individuelle virtuelle Sensoren eingebunden werden, die jeweils mit sämtlichen Zugriffsdaten zu versorgen sind. Durch den Einsatz in überschaubaren Anwendungsszenarien wurde auf modulare Konzepte verzichtet, die bei der Integration einer großen Zahl sehr heterogener Kontextinformationen jedoch notwendig sind. Auch fehlen Methoden zur anwendungsübergreifenden Modellierung von Kontextinformationen etwa durch synonyme Bezeichnung von Realweltobjekten oder die Unterstützung mehrerer Ontologien sowie die Modellierung von Kontextwissen durch Einbindung von Transformations- und Interpretationsmethodik in das Kontext-

Kontextinformation	Beschreibung
Aktivität	gerade ausgeführte Tätigkeit
Zeit	gegenwärtige Uhrzeit
Ort	gegenwärtige Position des Endgerätes
Identität	aktueller Benutzer des Endgerätes
Kapazität	verfügbare Speicherplatz auf dem Endgerät

Tabelle 16: Von Gaia ausgewertete Kontextinformationen

modell. Durch seinen sehr code-orientierten Ansatz kann das Kontextmodell des *Context Framework* auch nicht als endnutzertauglich bezeichnet werden.

4.6.8 Gaia

RANGANATHAN und CAMPBELL modellieren kontextbezogene Informationen im *Gaia*-Framework in Form von Prädikatenlogik direkt in der Faktendatenbank eines regelbasierten Expertensystems, wodurch diese gleich durch das Expertensystem analysiert und ausgewertet werden können. Motivation für diesen Ansatz ist die Überlegung, die Mächtigkeit von Verfahren des maschinellen Lernens zur Handhabung von Kontextsituationen und -daten, etwa zur Erkennung und Interpretation von Situationen zu verwenden (vgl. [Rom⁺02, RaCa03]). Das *Gaia*-Framework macht Informationen abhängig von der aktuellen Situation verfügbar, indem die in Tab. 16 dargestellten Kontextinformationen ausgewertet werden.

Grundlage des Kontextmodells stellen explizit in *DAML+OIL* dargestellte Ontologien dar (vgl. [RaCa03, S. 355], [Horr02]). Das Kontextmodell wird in einer Logiksprache, etwa *Prolog*, formuliert, wobei Kontextinformationen als *Fakten* und Auswertungsalgorithmen entsprechend als *Regeln* dargestellt werden. Sämtliche Daten werden mit einem Zeitstempel versehen in einer Datenbank persistent gespeichert. Zum Schlußfolgern von Kontextverhältnissen, etwa zum Bestimmen der Gültigkeit bestimmter Situationen stehen die aus der KI bekannten Inferenzmechanismen zur Verfügung (vgl. [RaCa03, S. 354]). Auftretende Konflikte durch widersprüchliche Regeln werden dadurch aufgelöst, dass die Regel mit der höchsten Priorität verwendet wird (vgl. [RaCa03, S. 360]). Höherwertige Ansichten werden aus existierenden Kontextinformationen durch *Synthesizer* genannte Interpreter erzeugt, welche diese Informationen jedoch nicht wieder in die Datenbasis eingliedern, sondern Client-Anwendungen direkt zur Verfügung stellen (vgl. Abb. 56 auf der gegenüberliegenden Seite). Weitere Verarbeitungsmöglichkeiten sind durch *Data Mining*-Verfahren gegeben.

Das verwendete Kontextmodell ist eindeutig, wohldefiniert, ausdrucksstark und erlaubt umfangreiche, auch sehr komplexe induktive und deduktive Schlußfolgerungsmöglichkeiten im Rahmen der Möglichkeiten von Expertensystemen (vgl. [RaCa03, Shmu87]). Es ist technisch gut handhabbar und kann aufgrund der Ausrichtung auf pervasive Anwendungsszenarien mit multidimensionalen, dynamischen und mit entsprechenden Erweiterungen auch mit qualitativ geringwertigen Daten umgehen.

Deutliche Lücken weist dieses Modell jedoch bei der Integration heterogener Informationen und der Endnutzertauglichkeit auf. Obwohl es zur Lösung beider in Kap. 1.2.2 („Problemstellung“, S. 8) beschriebenen Probleme eingesetzt werden kann, wurde es offenkundig nicht dafür konzipiert und kann

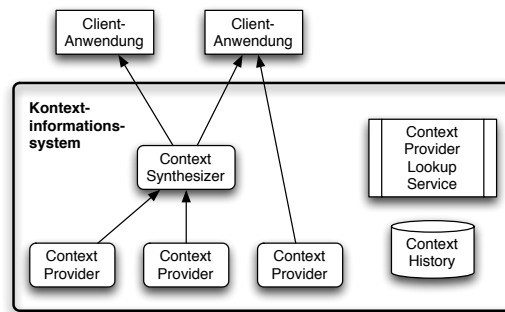


Abbildung 56: Integrationsstruktur des *Gaia*-Frameworks

auch nur mit fundamentalen Erweiterungen daran angepasst werden. Darüber hinaus sind die zur Benutzung von Expertensystemen verwendeten prädikatenlogischen Ausdrücke für Endnutzer unanschaulich (vgl. Tab. 17), der ganze Ansatz ist nichtmodular, unterstützt keine Abgrenzung der Gültigkeit von Aussagen auf einzelne Regionen und erlaubt keine anwendungsspezifischen Sichtweisen auf Teile der Datenbasis.

Anwendung finden können diese Systeme nur innerhalb relativ stark abgegrenzter, formalisierter und durchstrukturierter Anwendungsdomänen ohne widersprüchliche oder fehlende Aussagen, was jedoch gerade bei der Kontextverarbeitung nicht vorausgesetzt werden kann, da dort Situationen aus einem sehr großen Spektrum von Möglichkeiten auftreten können (vgl. [Pupp88]). Selbst innerhalb normaler betrieblicher Anwendungssysteme, welche i. d. R. auf eine strukturiertere Informationsbasis mit einer wesentlich höheren Datenqualität als kontextsensitive Anwendungssysteme zurückgreifen können, sind Expertensysteme, obwohl sie noch Ende der 1980er Jahre intensiv beforscht wurden, bis Mitte der 1990er Jahre fast vollständig verschwunden (vgl. [Sied99, S. 397], [Mert90, Mer⁺93, Kön⁺95]). Dies findet seine Ursache letztlich darin, dass die durch Expertensysteme bereitgestellte *mathematische* Formalität in der betriebswirtschaftlichen Praxis keine so dominierende Rolle wie in der Mathematik oder Informatik

Modell-element	Statement	Bedeutung
Ereignis	Location(chris, entering, room 3231)	„Chris“ betritt gerade „Raum 3231“
Aussage	Temperature (room 3231, '=', 24)	Die Temperatur in „Raum 3231“ beträgt 24 °C
Ontologie	SubClass(car, machine)	Auto ist eine Subklasse von Maschine
Abfrage	ALL People x (Location x; In; Room 3231)	alle Personen, die sich in Raum 3231 befinden
Regel	Social Activity (Room 3234, Party) :- Sound (Room 3234, '>', 40 dB) AND Lighting (Room 3234, Stroboscopic) AND NrPeople (Room 3234, '>', 6)	Wenn es in Raum 3231 lauter als 40 dB ist, stroboskopisches Licht herrscht, und sich mehr als 6 Personen im Zimmer befinden, findet dort eine Party statt.

Tabelle 17: Beispiele für verschiedene bei der Benutzung des Kontextmodells von *Gaia* verwendete Aussagen in Prädikatenlogik

Feld	Beschreibung
ID	ordnet den Tupel einem Realweltobjektmodell zu
Beschreibung	textuelle oder formale Beschreibung dieses Tupels
Einheit	Maßeinheit des enthaltenen Wertes
Wertebereich	Ergebnisraum für die enthaltenen Werte
Wert	Feld für den konkreten Messwert
Wahrscheinlichkeit	Wahrscheinlichkeit für die Gültigkeit dieses Messwertes

Tabelle 18: Tupel zur Beschreibung einer Kontextvariablen im Kontextmodell von *Schmidt*

spielt, da sie in eine ganze Reihe weiterer Optimierungsziele, in der vor allem die Wirtschaftlichkeit und Handhabbarkeit der entwickelten Lösungen hervorzuheben ist, einzuordnen ist (vgl. [Mess99, S. 109]).

4.6.9 Tupelraumbasierte Ansätze

SCHMIDT modelliert Kontext mittels eines *Tupelraum*-Ansatzes vor allem hinsichtlich der Darstellung und Verteilung sensorisch erfasster Kontextinformationen gemäß ihrer raumzeitlichen Relevanz (vgl. [ScGe01]). Diese Arbeit steht stellvertretend für eine Reihe weiterer tupelraumbasierter Kontextmodelle, die ähnliche Eigenschaften aufweisen, etwa den *MeLog*-Ansatz von MEISSEN (vgl. [Mei⁺04]).

Dabei werden vor allem orts- und zeitbasierte Kontextwerte durch in Tab. 18 dargestellte Tupel repräsentiert, die in einer Datenbank gespeichert werden. Das konkrete Format der Messwerte ist dabei nicht festgelegt und kann Skalare, Vektoren oder Zeichenketten umfassen. Erfassungsort und -zeit stellen dabei separat dargestellte Metainformationen zu den Kontexttupeln dar. Höherwertige Kontextsichtweisen, vor allem Entscheidungen über die Gegebenheit bestimmter Situationen, werden durch *Abstraktoren* realisiert, die basierend auf der Analyse der Datenbasis die betreffenden Informationen direkt an Clientanwendungen ausgeben (vgl. Abb. 33 auf Seite 99, [ScGe01, S. 9], [Sch⁺99a, S. 93]). Recherchen werden assoziativ mittels Suchschablonen im selben Tupelformat durchgeführt, bei denen einzelne Felder ausgefüllt werden, wobei die Ergebnismenge diejenigen Tupel umfasst, die der Suchschablone entsprechen.

Lokalität, d. h. die mit zunehmendem zeitlichen und räumlichen Abstand von ihrem Erfassungszeitpunkt und -ort abnehmende Relevanz von Kontextinformationen wird durch SCHMIDT durch die aus der *Fuzzy-Logik* bekannten trapez- oder dreiecksförmigen Relevanzfunktionen modelliert, weswegen dessen Kontextmodell auch als *FuzzySpace* bezeichnet wird (vgl. [ScGe01, S. 5 ff]). Dadurch müssen jeder Clientanwendung nur die als für sie relevanten Kontextinformationen, d. h. die aus ihrer zeitlichen oder räumlichen Nähe, bereitgestellt werden, wodurch die Menge der von den Clientanwendungen zu verarbeitenden Kontextinformationen drastisch reduziert werden kann.

Das auf einem Tupelraumkonzept basierende Kontextmodell von SCHMIDT dient vor allem der Organisation der im *Ubiquitous Computing* sehr zahlreich anfallenden Kontextinformationen, weswegen seine größte Stärke in der automatischen Einengung der Sichtweise auf die räumlich oder zeitlich naheliegenden Kontextinformationen sowie der automatisiert möglichen Bereinigung der Kontextinformationsbasis von irrelevant gewordenen Informationen liegt. Hinsichtlich der in dieser Arbeit gegebenen

Problemstellung kann eine gute Unterstützung dynamischer Daten sowie eine nur bedingte Eignung zur Integration heterogener Kontextinformationen festgestellt werden, wobei gerade dieser Aspekt vom Kontextinformationsmodell konzeptuell nicht unterstützt wird.

Das Kontextmodell ist also vor allem hinsichtlich der räumlichen und zeitlichen Diskriminierung von Kontextinformationen optimiert. Eine Integration heterogener Kontextinformationen würde nicht inhaltlich, sondern über die räumliche Nähe zu anderen Kontextinformationen hergestellt werden. Insgesamt stellt dieses Modell keine geeignete Grundlage für ein auf die semantische Integration heterogener Kontextinformationen optimiertes Kontextmodell dar.

4.6.10 Auswertung

Die Entwicklung im Bereich der Modellierung von Kontext hat zu einer Vielzahl allgemein gut handhabbarer Kontextmodelle geführt, wobei aus jeder Kategorie von Informationsstrukturierungsarten ein exemplarisches Beispiel vorgestellt und hinsichtlich der für diese Arbeit relevanten Kriterien bewertet wurde. Dabei konnte gezeigt werden, dass bestimmte Ansätze durchaus in der Lage sind, einzelne Anforderungen hinsichtlich der Darstellung integrierter Kontextinformationen angemessen zu erfüllen, jedoch weisen sämtliche verwandten Arbeiten durch die fehlende prinzipielle Orientierung auf die gegebene Problemstellung durchaus gravierende, oft nicht ohne fundamentale Modifikationen überwindbare Lücken auf.

Abb. 57 auf der nächsten Seite stellt die Ergebnisse der Anforderungsanalyse im Detail und Abb. 58 auf Seite 133 zusammengefasst dar. Die Graustufen korrelieren dabei mit der Bewertung und sollen die Regionen, in denen eine besonders gute bzw. schlechte Bewertung erfolgte, visualisieren. Die Spalte *Context Maps* stellt die Bewertung des in dieser Arbeit entwickelten Kontextmodells dar und soll an dieser Stelle noch nicht interessieren. Details zur Bewertung der einzelnen Arbeiten finden sich wie bereits erwähnt in Anhang A („Detaillierte Bewertungen verwandter Arbeiten“, S. XXXVII).

Überwiegend gut fällt die technische Handhabbarkeit der bereits bestehenden Lösungen aus, was sich i. d. R. jedoch nicht zwangsläufig in einer befriedigenden Endnutzertauglichkeit niederschlägt. Ebenfalls als gelöst angesehen werden können die multidimensionalen sowie dynamischen Anforderungskomplexe. Datenqualität und -sicherheit spielen in heutigen Kontextmodellen zumeist keine Rolle, trotzdem kann hier auf bereits hinreichende Lösungen verwiesen werden.

Im Bereich der Integration heterogener Kontextinformationen können zufriedenstellend gelöste sowie bisher noch nicht adressierte Anforderungen identifiziert werden. Die meisten der dargestellten Verfahren erlauben durch Nutzung verschiedener Arten von *Software-Sensoren* die Einbindung beliebiger externer Informationen in das Kontextmodell. Dies erfolgt jedoch auf einer zu unspezialisierten Basis. Die Adressierung konkreter Datenfelder in externen Kontextquellen wird nicht zufriedenstellend angegangen. Durchgängig fehlt eine zur Beschaffung mehrerer Kontextvariablen wiederverwendbare Repräsentation von externen Kontextquellen, welche etwa Zugangsdaten zur Verwendung durch mehrere Softwaresensoren bereitstellt.

Auch das Hantieren mit verschiedenen, z. T. anwendungsspezifischen Ontologien und Sichtweisen auf das Kontextmodell ist mit den existierenden Arbeiten nur sehr bedingt bzw. umständlich lösbar. Die Einengung der Sichtbarkeit von Konzepten auf bestimmte Zusammenhänge ist so gut wie nie möglich,

		rel. DB (ERM)	OODB	XML	semWeb/RDF	Power (Topic Maps)	Context Cube	Henricksen	Context Toolkit	Context Fabric	Sentient Computing	Context Framework	Gaia	Schmidt	Context Maps
Gesamtwertung		2,0	3,0	2,9	2,9	3,0	3,1	2,7	3,1	3,8	3,2	2,9	3,1	3,2	4,9
1	Allgemeine Verwendbarkeit	3,0	3,6	3,8	4,0	4,0	3,0	3,0	3,0	4,0	3,6	2,6	4,0	3,2	5,0
1a	Formalisierungsgrad	4	4	5	5	5	5	2	4	5	5	4	5	3	5
1b	Modularität	1	5	5	5	5	1	2	3	1	4	1	1	1	5
1c	Verteilte Darstellung	1	4	1	1	1	1	2	5	4	4	3	4	4	5
1d	Programm.sprachenunabh.	5	1	5	5	5	5	5	1	5	1	1	5	5	5
1e	flex. Addressierungsmögl.	4	4	3	4	4	3	4	2	5	4	4	5	3	5
2	Multidimensionalität	2,0	5,0	5,0	5,0	5,0	5,0	2,0	5,0	4,0	5,0	3,0	4,5	4,5	5,0
2a	Separate Datendimensionen	3	5	5	5	5	5	2	5	5	5	5	5	5	5
2b	Kombination v. Dimensionen	1	5	5	5	5	5	2	5	3	5	1	4	4	5
3	Dynamik	1,7	4,3	2,7	2,7	2,3	3,7	2,7	3,0	4,0	4,0	3,3	4,7	3,7	5,0
3a	Dynamische Elemente	2	4	1	1	1	1	4	3	3	5	4	4	3	5
3b	generalisierende Ontologien	1	5	5	5	5	5	1	1	4	5	4	5	3	5
3c	Historie	2	4	2	2	1	5	3	5	5	2	2	5	5	5
4	Datenqualität	1,5	2	2	2	1	2	5	2	3,5	2,5	5	2	3,5	5
4a	Qualitätsdeskriptoren vorh.	2	2	2	2	1	2	5	2	5	3	5	2	5	5
4b	datentypspez. Qualitätsdeskr.	1	2	2	2	1	2	5	2	2	2	5	2	2	5
5	Datensicherheit	1,3	2,0	2,0	2,0	2,0	2,0	2,0	2,0	5,0	2,0	2,0	2,0	2,0	4,3
5a	Datenschutzdeskriptoren	2	2	2	2	2	2	2	2	5	2	2	2	2	5
5b	Darstellung von Akteuren	1	2	2	2	2	2	2	2	5	2	2	2	2	4
5c	Vertrauensstufen	1	2	2	2	2	2	2	2	5	2	2	2	2	4
6	Integration heterogener CI	2,1	3,2	2,5	2,6	3,2	3,4	2,9	3,1	3,8	3,4	2,6	2,6	3,1	5,0
6a	Stellvertreteransatz	5	5	5	4	5	5	5	4	5	5	5	1	3	5
6b	eindeutige RWOM	2	2	2	5	5	3	4	2	2	1	4	1	3	5
6c	explizite Ontologien	3	4	5	5	5	5	2	1	5	4	4	5	4	5
6d	Kontextquellen	1	2	2	2	2	5	2	2	4	2	2	4	3	5
6e	Kontextvariableninhalte	3	4	3	3	1	5	4	5	5	5	4	5	5	5
6f	Links auf Originaldaten	1	2	4	2	4	4	1	4	4	1	4	1	1	5
6g	Fusion von Kontextvariablen	1	2	1	1	1	4	3	4	3	4	1	4	2	5
6h	Integration von Methodik	1	5	1	1	1	3	3	4	3	5	1	4	4	5
6i	Darstellung des Datenpfades	3	4	1	1	1	1	5	5	5	5	1	1	4	5
6j	Sichtbarkeit einschränken	1	3	1	3	5	1	1	1	1	3	1	1	4	5
6k	Synonyme Objektnamen	2	2	2	2	5	1	2	2	5	2	2	2	1	5
7	Endnutzertauglichkeit	2,7	1,0	2,3	2,3	3,7	2,3	1,3	3,7	2,3	1,7	1,7	2,0	2,3	4,3
7a	versch. Abstraktionsstufen	1	1	2	2	2	3	2	3	1	1	1	3	3	5
7b	endnutzertaugl. Darstellung	3	1	3	3	4	3	1	4	4	3	3	1	3	4
7c	subjektive Sichtweisen	4	1	2	2	5	1	1	4	2	1	1	2	1	4

Abbildung 57: Überblick über die Bewertung der in dieser Arbeit diskutierten Arbeiten zur Integration und Darstellung verteilter und heterogener Kontextinformationen

		rel. DB (ERM)	OODB	XML	semWeb/RDF	Power (Topic Maps)	Context Cube	Henricksen	Context Toolkit	Context Fabric	Sentient Computing	Context Framework	Gaia	Schmidt	Context Maps
1	Allgemeine Verwendbarkeit	3,0	3,6	3,8	4,0	4,0	3,0	3,0	3,0	4,0	3,6	2,6	4,0	3,2	5,0
2	Multidimensionalität	2,0	5,0	5,0	5,0	5,0	5,0	2,0	5,0	4,0	5,0	3,0	4,5	4,5	5,0
3	Dynamik	1,7	4,3	2,7	2,7	2,3	3,7	2,7	3,0	4,0	4,0	3,3	4,7	3,7	5,0
4	Datenqualität	1,5	2,0	2,0	2,0	1,0	2,0	5,0	2,0	3,5	2,5	5,0	2,0	3,5	5,0
5	Datensicherheit	1,3	2,0	2,0	2,0	2,0	2,0	2,0	2,0	5,0	2,0	2,0	2,0	2,0	4,3
6	Integration heterogener CI	2,1	3,2	2,5	2,6	3,2	3,4	2,9	3,1	3,8	3,4	2,6	2,6	3,1	5,0
7	Endnutzertauglichkeit	2,7	1,0	2,3	2,3	3,7	2,3	1,3	3,7	2,3	1,7	1,7	2,0	2,3	4,3

Abbildung 58: Durchschnittliche Bewertung der Anforderungskomplexe

was jedoch aus datenschutztechnischen als auch ergonomischen Gesichtspunkten wünschenswert wäre, da hierdurch der Zugriff auf semantisch verschiedene Bereiche der Datenbasis flexibel eingeschränkt werden kann. Ebenso stellen die Unterscheidung sowie Kombination synonyme Kontextvariablen in unterschiedlichen Kontextquellen sowie die Identifikation von Konzepten mittels synonyme Identifikatoren bislang in Kontextmodellen überhaupt nicht adressierte Problemstellungen dar. Eineindeutige Realweltobjektabbildungen werden bei allen Modellen zwar gefordert, es werden jedoch keine weiteren Mittel bereitgestellt, um dies zu formalisieren, so dass diese Forderung von entsprechenden Anwendungen nicht formalisiert unterstützt werden kann.

Methodische Elemente werden entweder durch außerhalb der Kontextinformationsbasis befindliche externe Agenten oder durch objektorientierte Verfahren dargestellt, wobei beide Konzepte für die gegebene Problemstellung Grenzen besitzen (vgl. Kap. 4.3 „Integration aktiver Elemente in Datenstrukturen“, S. 108)). Damit zusammenhängend kann der Datenpfad zwar durchaus von einer Reihe von verwandten Arbeiten dargestellt werden, was jedoch nicht besonders übersichtlich bzw. intuitiv erfolgt.

Vollständig endnutzertauglich ist keines der dargestellten Verfahren, wobei hier im Besonderen unterschiedlich abstrakte bzw. detaillierte Abbildungen sowie Darstellungen der Informationsbasis mittels verschiedener Ontologien fehlen. Auch mangelt es vielen Modellen an Intuitivität, d. h. sie sind für einen untrainierten Endnutzer nicht einsichtig, sondern können erst nach mehr oder weniger umfassenden Fortbildungen verwendet werden.

Abb. 59 auf der nächsten Seite stellt die jeweils schlechteste Bewertung für jeden Anforderungskomplex dar und belegt dadurch, dass jedes der dargestellten Verfahren in fast allen Anforderungskomplexen gravierende Lücken aufweist und damit nicht zur umfassenden Integration von Kontextinformationen taugt. Dies gilt insbesondere hinsichtlich der Eignung zur Integration von heterogenen Kontextinformationen, was durch Zeile 6 in Abb. 59, die besonders viele mit nur einem Punkt und damit nicht erfüllbare Anforderungen enthält, verdeutlicht wird. Als Folge davon müssen diese Anforderungen durch proprietäre, nicht standardkonforme Funktionalitäten realisiert werden, was wiederum die Austauschbarkeit und Integrationsfähigkeit dieser Lösungen erschwert, wodurch die Vorteile der Nutzung existierender Standards nicht zum Tragen kommen können. Daher ist die Entwicklung

	rel. DB (ERM)	OODB	XML	senWeb/RDF	Power (Topic Maps)	Context Cube	Henricksen	Context Toolkit	Context Fabric	Sentient Computing	Context Framework	Gaia	Schmidt	Context Maps
1	Allgemeine Verwendbarkeit	1	1	1	1	1	2	1	1	1	1	1	1	5
2	Multidimensionalität	1	5	5	5	5	2	5	3	5	1	4	4	5
3	Dynamik	1	1	1	1	1	1	1	3	2	2	4	3	5
4	Datenqualität	1	2	2	2	1	2	5	2	2	5	2	2	5
5	Datensicherheit	1	2	2	2	2	2	2	5	2	2	2	2	4
6	Integration heterogener CI	1	2	1	1	1	1	1	1	1	1	1	1	5
7	Endnutzertauglichkeit	1	1	2	2	2	1	1	3	1	1	1	1	4

Abbildung 59: Minima bei der Bewertung der Anforderungskomplexe

einer eigenen Lösung für dieses Problem, die für sämtliche angesprochenen Anforderungen passende Lösungen bereithält, sowohl gerechtfertigt als auch motiviert. Daher wird im nächsten Schritt ein neues Kontextinformationsintegrations- und -darstellungsmetamodell entwickelt, welches Verfahren für die noch offenen Problemstellungen beinhaltet und somit allgemein als erstes Metamodell *sämtliche* der dargestellten Anforderungen erfüllt und damit die erste umfassend brauchbare Lösung für die in dieser Arbeit entwickelte Problemstellung darstellt.

4.7 Zusammenfassung

In diesem Kapitel wurden die wichtigsten verwandten Arbeiten aus dem Bereich der Integration und Darstellung verteilter und heterogener Kontextinformationen vorgestellt. Zunächst wurden die wichtigsten in Kontextquellen verwendeten Kontextdatenformate vorgestellt, anschließend Integrationsansätze für heterogene Informationsbestände sowie für methodische Elemente in Datenstrukturen. Nach der Bewertung existierender Informationsstrukturierungsmetamodelle wurden spezielle Darstellungsmetamodelle für Kontextinformationen hinsichtlich ihrer Tauglichkeit für die gegebene Problemstellung untersucht.

Keine der dargestellten verwandten Arbeiten kann sämtliche Anforderungen insbesondere hinsichtlich der Integration von kontextbezogenen Informationen aus verschiedenen Kontextquellen erfüllen, weswegen Realisierungen dieses Vorhabens auf vorhersehbare Probleme stoßen werden. Daher wird im nächsten Kapitel ein eigenes Kontextmodell entwickelt, das insbesondere diesen Schwerpunkt adressiert.

5 Context-Maps

***Im Grunde sind die meisten benötigten
Informationen bereits irgendwo vorhanden.
Das Problem ist, dass wir dies nicht berücksichtigen
und daher nicht wissen,
was wir eigentlich alles wissen.***

*Prof. Dr. Eric Schoop,
TU-Dresden*

Dieses Kapitel stellt einen Ansatz zur strukturellen und inhaltlichen Integration der Inhalte von verteilten und heterogenen Kontextquellen vor. Dazu wird die Problemdomäne in den Kap. 5.1.2 bis 5.5 zunächst aus einer informations- und modelltheoretischen Perspektive formalisiert, analysiert und die Lösung mit theoretischen Mitteln beschrieben. Darauf aufbauend werden ab dem Kap. 5.4 *Context Maps*, ein diese Prinzipien realisierendes und in einem wirtschaftsinformatischen Sinn praktisch einsetzbares Metamodell zur Darstellung von Kontextmodellen, die durch Kombination heterogener, aus verschiedenen Datenquellen zusammengetragenen kontextbezogenen Informationen bestehen, vorgestellt.

5.1 Informationsintegrationsmodell

Nach der in Kap. 2 erfolgten grundlegenden Einordnung und Analyse des Problems wird an dieser Stelle eine informationstheoretische Lösung für die speziell in dieser Arbeit betrachtete Problemstellung skizziert. Aufgrund der Komplexität und Vielschichtigkeit des Problembereiches werden dazu diverse Modelle verwendet, die jeweils gezielt einen Teilaspekt modellieren.

Entsprechend den beiden in Kap. 1.2.2 („Problemstellung“, S. 8) beschriebenen Problemen werden die Teilbereiche Informationszugriff, Informationsintegration auf Realweltobjektmodellebene sowie eine Integration der verschiedenen Realweltobjektmodelle zu einem Realweltmodell jeweils separat gelöst und diese Integrationsteilmodelle später miteinander zu einem umfassenden Integrationsmodell kombiniert. Diese drei Teilmodelle werden im folgenden vorgestellt.

5.1.1 Zugriffsmodell

Das *Zugriffsmodell* stellt die in den heterogenen Kontextquellen verteilten Kontextinformationen in einer einheitlichen Weise in Form von *Kontextvariablen* dar und macht sie so einheitlich und technisch transparent zugreifbar. Dies wird in Abb. 60 auf der nächsten Seite visualisiert.

Zur Identifikation bzw. zum Zugriff der betreffenden externen Kontextinformationen werden pro Kontextvariable jeweils die in Tab. 19 auf der nächsten Seite dargestellten Informationen benötigt. Die Informationen über die Datenquelle, d. h. deren Adresse, Zugriffs- und Loginverfahren werden dabei in einer separaten Form in einem *Quellenmodell* gespeichert. Dies erfolgt zum einen, um diese zur Adressierung mehrerer Kontextvariablen verwendbaren Informationen in einer nichtredundanten, d. h.

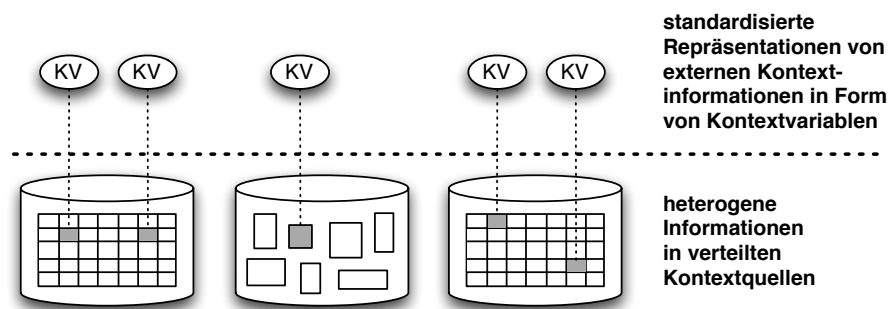


Abbildung 60: Visualisierung des Zugriffsmodells

wiederverwendbaren Form darzustellen, zum anderen, um zumindest die Login-Informationen für Kontextquellen nicht für jeden, jederzeit und überall sichtbar darstellen zu müssen, was eine grundlegende datenschutztechnische Anforderung, die von diesem Integrationsmodell auch konzeptionell zu berücksichtigen ist, darstellt. Anzumerken bleibt, dass das Zugriffsmodell i. d. R. nicht separat dargestellt, sondern zusammen mit dem nachfolgend beschriebenen Integrationsmodell verwendet wird, d. h. einzelne externe Kontextvariablen werden nicht separat, sondern nur *innerhalb* des im folgenden beschriebenen Integrationsmodells dargestellt.

5.1.2 Integrationsmodell

Mittels des *Integrationsmodells* werden die einzelnen, durch das *Zugriffsmodell* adressierten Kontextvariablen zu Realweltobjektmodellen zugeordnet. Dabei werden durchaus mehrere synonyme Kontextvariablen inhaltlich per Fusionsoperation zusammengefasst, um als *ein* Attribut eines Realweltobjektmodells dargestellt werden zu können.

Abb. 61 auf der gegenüberliegenden Seite stellt diesen Vorgang schematisch dar. Mittels eines beliebigen Integrationsalgorithmus, welchen das *IA* genannte Modellelement dargestellt oder referenziert, wird aus einer Reihe von nun nicht mehr heterogenen und verteilten, dafür aber immer noch isolierten Kontextvariablen der Wert eines Attributes eines im folgenden weiter besprochenen Realweltobjektmodelles bestimmt. Ein solches Realweltmodell repräsentiert ein konkretes Realweltobjekt.

Informationsart	Beschreibung
Quellentyp	der Typ der Quelle wird durch ein Kürzel angegeben, um daraus ein geeignetes Zugriffsverfahren auf die Quelle abzuleiten.
Logindaten	Informationen, um den Zugriff auf entsprechend gesicherte Kontextquellen zu autorisieren.
Adresse der Quelle	Informationen, wo die Quelle zu finden ist.
Lokalisierung der Kontextvariablen	Informationen darüber, wie die Kontextvariable <i>innerhalb</i> der angegebenen Kontextquelle adressiert werden kann.

Tabelle 19: Zum Zugriff auf externe Kontextvariablen benötigte Informationsarten

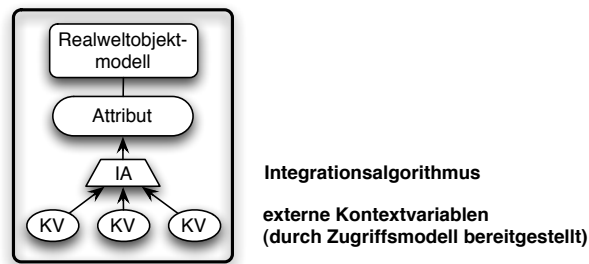


Abbildung 61: Modell zur Integration isolierter Kontextvariablen in einen Attributwert eines Realweltobjektes

5.1.3 Verbindungsmodell

Während das *Integrationsmodell* die Zusammenhänge zwischen sehr naheliegenden, d. h. zum selben Realweltobjekt zuordbaren Kontextvariablen modelliert, werden im *Verbindungsmodell* die indirekten, d. h. die Zusammenhänge zwischen Kontextvariablen verschiedener Realweltobjekte modelliert. Dazu werden die betreffenden Realweltobjektmodelle durch Darstellung der zwischen den entsprechenden Realweltobjekten geltenden Zusammenhänge miteinander assoziiert. Dies erfolgt mittels eines *semantisches Netzes*, in dem Realweltobjektmodelle als Knoten und die Verbindungen zwischen diesen als Kanten dargestellt werden. Knoten und Kanten können dabei basierend auf einer Ontologie beliebig typisiert sein, wodurch ihre Bedeutung semantisch aussagefähig und anwendungsübergreifend bekannt gemacht wird. Abb. 62 visualisiert ein typisches Verbindungsmodell. Zu sehen sind einige Realweltobjektmodelle samt Attributen, die durch fett dargestellte Kanten miteinander verbunden sind, wodurch letztlich sämtliche Kontextvariablen in einen umfassenden und ganzheitlichen Zusammenhang gebracht wurden.

5.1.4 Datenverteilungsaspekte

Die Datenhaltung bei *Context Maps* unterliegt wie in vorigen Kapiteln bereits angedeutet den im folgenden dargestellten Paradigmen.

Doppelte Datenhaltung. Aufgrund der in Anf. 6f („Verbindung vom Kontextmodell zu den Originaldaten“) beschriebenen und in Abb. 17 auf Seite 38 visualisierten doppelten Datenhaltung liegen die Kontextinformationen sowohl in originaler Form in den Kontextquellen als auch in aufbereiteter Form in den CONTEXT MAPS vor. Anwendungen steht es daher letztlich frei, die Kontextinformationen (a) in einer leicht zugreifbaren und vorverwerteten Form aus den *Context Maps* oder (b) performant direkt aus den

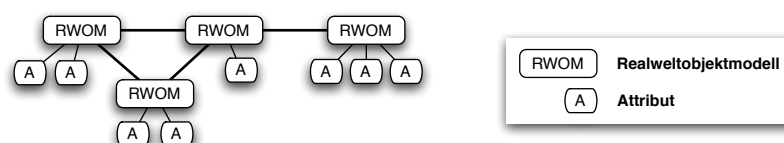


Abbildung 62: Schematische Darstellung des Verbindungsmodells

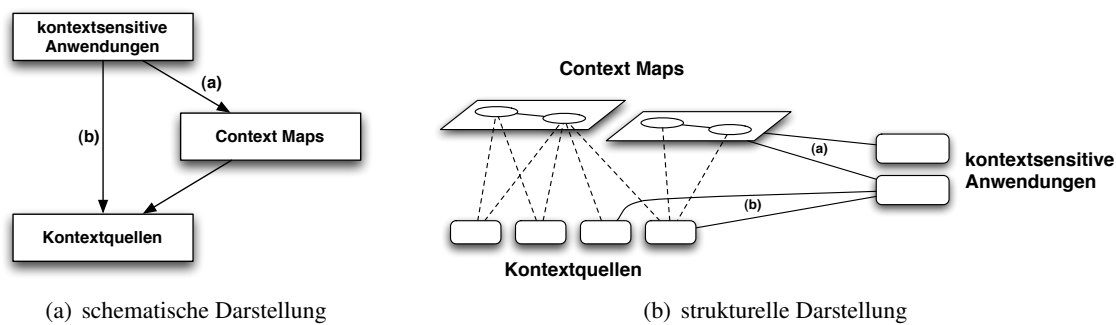


Abbildung 63: Zugriffsmöglichkeiten auf Kontextinformationen

Kontextquellen zu beziehen. Beide Verfahren können gleichzeitig angewendet werden, da sie sich nicht gegenseitig beeinflussen. Dies wird in Abb. 63 in schematischer sowie struktureller Form dargestellt.

Verteilte Datenhaltung. Um den Anforderungen einer verteilten und mobilen Umgebung gerecht zu werden, kann und soll es nicht nur eine zentrale Context Map geben, welche die auf den diversen mobilen Geräten anfallenden Kontextinformationen integriert, sondern die Informationen sollen relativ beliebig auf verschiedene Maps, die sich wiederum auf verschiedenen Geräten befinden können, verteilt werden können. Dabei werden Informationen in anderen Maps durch die zusätzliche Angabe des Namens der betreffenden Map sowie der Adresse des betreffenden Servers adressiert. Damit stellen Context Maps nicht nur eine einheitliche Sichtweise und Zugriffsschnittstelle auf heterogene Informationen aus verschiedenen Kontextquellen, sondern selbst wiederum ein allerdings transparent verteiltes Informationsmodell dar, welches sich an die gegebene Geräte- und Netzwerktopologie anpassen lässt und einen transparenten Zugriff auf die verteilten Informationen ermöglicht. Dies wird in Abb. 64 dargestellt, welche ein Beispiel für diverse auf verschiedene Geräte verteilte, aber untereinander verbundene Context Maps darstellt, die zusammen ein geräteübergreifendes Kontextmodell bilden.

Hierarchische Datenhaltung Das soeben dargestellte Prinzip, die Inhalte von Context Maps auf verschiedene Maps aufzuteilen, ermöglicht durch eine hierarchische Anordnung verschieden abstrakter Context Maps die effiziente Darstellung umfangreicher *high-level*-Kontextinformationen bei redundanzfreier Beschaffung und Darstellung der dazu benötigten *low-level*-Informationsbasis (vgl. Abb. 65 auf der gegenüberliegenden Seite). Dazu werden Informationen eines Problemgebietes nur *einmal* beschafft und in einer allgemein verwendbaren Form auf der untersten abgebildeten Ebene dargestellt. Diese *low-level*-Kontextinformationsbasis kann dann schrittweise zu *mehreren* anwendungs- bzw. problemspezifischen *high-level*-Kontextdarstellungen abstrahiert werden, wobei obere Ebenen ausschließlich Daten

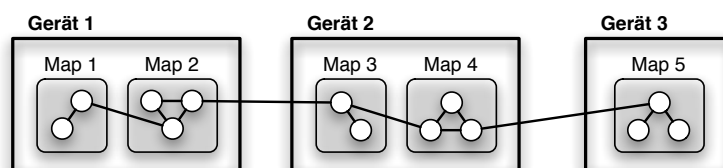


Abbildung 64: Auf diverse Geräte und Maps verteilte, aber miteinander verbundene Datenhaltung

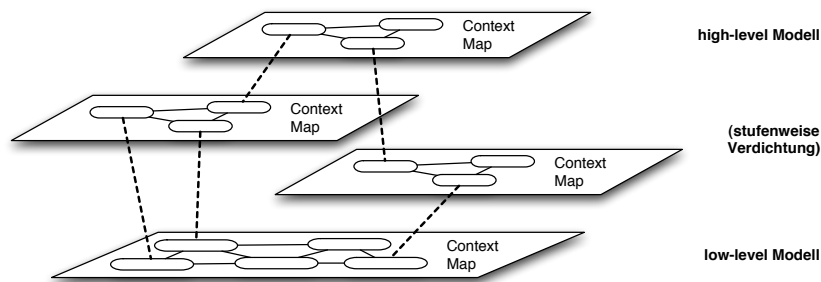


Abbildung 65: Stufenweise anwendungsspezifische Verdichtung der Datenbasis

aus unteren Ebenen beschaffen und verwenden (vgl. Abb. 19 auf Seite 46). Dadurch kann eine relativ komplexe und umfangreiche *low level*-Kontextinformationsbasis nicht nur schrittweise zu abstrakteren Sichtweisen verdichtet werden, wobei die inhaltlichen und strukturellen Beziehungen zwischen den jeweiligen *high level* und *low level* Informationen bestehen bleiben, sondern diese Informationsbasis kann unter Nutzung verschiedener Ontologien in verschiedenen anwendungsspezifischen Sichtweisen dargestellt werden (vgl. Abb. 19 auf Seite 46 und Abb. 20 auf Seite 46).

5.1.5 Gesamtarchitektur

Durch die Kombination sämtlicher der eben dargestellten Einzelmodelle entsteht ein komplexes Informationsintegrationsmodell, welches beliebige Informationen in verteilten und heterogenen Kontextquellen als Kontextvariablen darstellt und diese durch Fusion zu Attributwerten von miteinander assoziierten Realweltobjektmodellen integriert. Abb. 66 zeigt die resultierende Gesamtarchitektur. Dabei können sämtliche verwendeten Algorithmen und Metainformationen, d. h. die verschiedenen Fusionsalgorithmen und die Bedeutung der verschiedenen Informationsarten explizit dargestellt bzw. durch Ontologien semantisch aussagekräftig definiert werden. Das entstehende Informationsintegrationsmodell erlaubt die flexible Integration beliebiger Kontextinformationen aus verteilten und heterogenen Datenquellen.

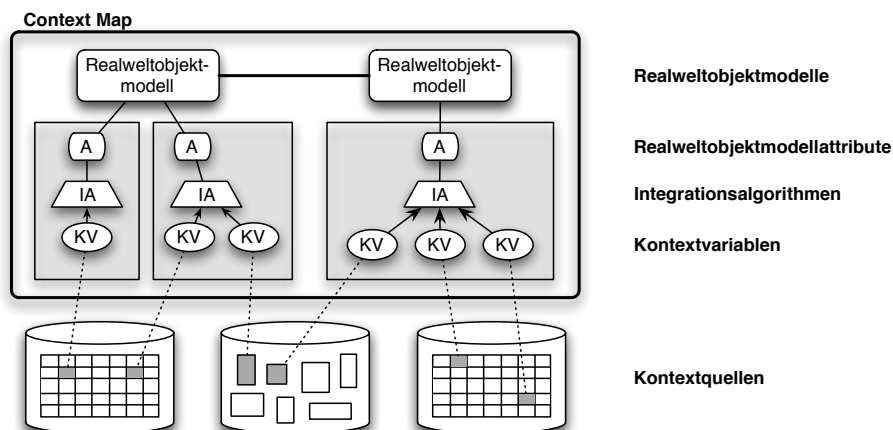


Abbildung 66: Gesamtes theoretisches Integrationsmodell

5.2 Kontextmodell

Nach der eben erfolgten Darstellung des Integrationsmodells für verteilte und heterogene Informationen wird an dieser Stelle etwas näher auf die Modellierung von Kontext eingegangen, was wiederum in Form diverser später zu einem Gesamtmodell kombinierter Teilmodelle, welche separate Aspekte der Kontextmodellierung beschreiben, erfolgt.

5.2.1 Grundlegende Kontextmodellelemente

Die Kontextmodellierung basiert auf dem bereits in Kap. 4.1.1 auf S. 97 vorgestellten *Stellvertreteransatz*, welcher Kontextinformationen durch Zuordnung zu Repräsentationen von Realweltobjekten organisiert. Hierdurch werden die Kontextvariablen nicht nur analog zu den in der Realwelt existierenden Objektstrukturen und damit für einen Nutzer mit Domänenwissen intuitiv organisiert, da Nutzer mit Domänenwissen die Struktur, Benennung und Organisation der Realwelt im Kontextmodell wiedererkennen und sich so intuitiv darin zurechtfinden können, vielmehr werden dadurch auch die unterschiedlichen *Zusammenhänge* zwischen den Kontextvariablen in angemessener Weise abgebildet, indem *synonyme* Kontextvariablen zu einem Realweltobjektmodell-Attribut fusioniert, *thematisch zusammengehörige* Kontextvariablen zum selben Realweltobjektmodell zugeordnet werden, während die indirekten Zusammenhänge zwischen *relativ unabhängigen* Kontextvariablen durch die ggf. erst über mehrere Zwischenschritte modellierten Zusammenhänge ihrer jeweiligen Realweltobjektmodelle dargestellt werden (vgl. Abb. 62 auf Seite 137).

Kontextmodelle bestehen entsprechend aus Realweltobjektmodellen, die durch Assoziationen untereinander verbunden sind und denen Kontextvariablen in Form von Attributen zugeordnet werden (vgl. Kap. 5.1.2 („Integrationsmodell“, S. 136)).

Um die in praxisnahen Kontextmodellen vorhandene Komplexität, die tausende solcher Modellelemente enthalten kann, angemessen handhaben zu können, werden im folgenden komplexere Beschreibungselemente für Kontext dargestellt.

5.2.2 Universen

Unter einem *Universum* wird ein subjektives Weltmodell verstanden, welches vor allem eine subjektive kontextbezogene Ontologie, d.h. Terminologie und Interpretation von in dieser Welt auftretenden Objekttypen und -vorgängen, aber auch zusätzliches Bedeutungs- und Problemlösungswissen, etwa die in dieser Domäne relevanten Eigenschaften, Fähigkeiten, Rollen, Aufgaben und Zielsetzungen der verschiedenen Charaktere enthält. *Subjektiv* bedeutet hierbei die Interpretation der Realwelt aus dem Blickwinkel einer Anwendung bzw. von deren Programmierer aus und ist begründet in der prinzipiellen Unmöglichkeit, die Realwelt objektiv zu repräsentieren.

Der Name für Universen wurde der phantastischen Literatur entlehnt, in welcher ein Universum ebenfalls das zum Verständnis einer Romanreihe notwendige Hintergrundwissen, etwa hinsichtlich der Eigenschaften, Fähigkeiten und Ziele von Charakteren und Objekten beinhaltet. Beispiele für Universen in der Literatur stellen das *Herr der Ringe*-Universum von TOLKIEN oder das im wissenschaftlichen Kontext beliebtere *Hitchhiker*-Universum von ADAMS dar (vgl. [Tolk54], [Adam79]). Ohne die Kenntnis dieses

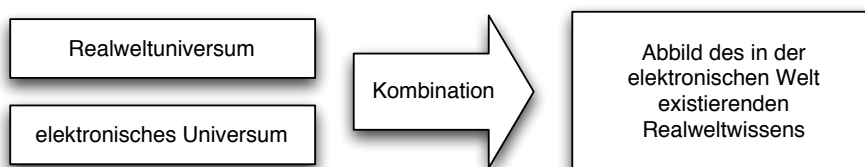


Abbildung 67: Kombination von elektronischen mit Realweltuniversen zu elektronischen Realweltmodellen

Hintergrund- oder Domänenwissens ist es nicht möglich, eine einzelne Episode aus den betreffenden Romanreihen bzw. übertragen auf diese Arbeit einen konkreten Vorgang in der Realwelt angemessen zu interpretieren. Dies soll durch zwei Beispiele für ein literarisches und kontextbezogenes Universum illustriert werden.

Beispiel 13 *Die Szene, in welcher ein Polizeiraumschiff nach einem Gespräch mit dem Roboter Marvin in die Luft fliegt wird erst verständlich, wenn bekannt ist, dass Marvin im Hitchhiker-Universum ein hochintelligenter, aber ständig unterforderter und damit depressiver Roboter ist, der alles in grotesker Weise infrage stellt und negativ interpretiert und das Raumschiff dadurch dazu brachte, Selbstmord zu begehen (vgl. [Adam79, S. 228f]).*

Beispiel 14 *Um die Fahrtziele eines Nutzers zu bestimmen, ist es notwendig, zu wissen dass dieser ein Mensch ist und Menschen sich ihre Termine in einem Kalender notieren sowie per EMail miteinander kommunizieren, so dass diese beiden Informationsquellen auf Hinweise nach möglichen Fahrtzielen durchsucht werden können. Ist dagegen bekannt, dass der Nutzer ein Angestellter einer Speditionsfirma ist, so sollte zuerst deren Logistiksystem nach Transportaufträgen für den aktuellen Nutzer befragt werden.*

In der Literatur, wo die möglichst spannende und unterhaltsame Entdeckung dieses Universums eine wichtige Rolle spielt, wird das Universum i. d. R. nicht explizit formuliert, sondern implizit in den einzelnen Episoden aufgebaut. Im wissenschaftlich-technischen Umfeld, wo es vor allem auf die möglichst effektive Nutzung dieser Informationen durch verschiedene Instanzen ankommt, werden jedoch eindeutige, formalisierte Repräsentationen von Domänenwissen und damit explizite Universen benötigt.

Entsprechend dem Ansatz dieser Arbeit, die Informationen aus der elektronischen Welt in geeigneter Weise zu Modellen der Realwelt zu kombinieren, können mit Realweltuniversen und elektronischen Universen zwei unterschiedliche Arten von Universen unterschieden werden, welche jeweils die Real- bzw. elektronische Welt beschreiben. Entsprechend stellt diese Arbeit durch die Kombination elektronischer mit Realweltuniversen eine Methode zur Synthese strukturierter Realweltmodelle aus heterogenen Kontextinformationen dar (vgl. Abb. 67). Nähere Informationen zu diesem Ansatz werden im weiteren Verlauf dieses Kapitels gegeben.

Um umfassende Sichtweisen auf den Kontext durch Kombination verschiedener anwendungsspezifischer Sichtweisen (d. h. Universen) zu erschaffen, wird eine Überführung der mittels verschiedener Universen dargestellten Informationen notwendig. Um diese automatisiert durchführen zu können, werden interontologische bzw. interuniversäre Konzepte benötigt, auf die in dieser Arbeit unter Verweis

auf die Arbeiten etwa von MENA jedoch nicht genauer eingegangen werden soll. Vielmehr stellt diese Arbeit ein Darstellungsmodell für Kontextinformationen zur Verfügung, welches als Resultat entweder einer manuellen oder automatischen Universentransformation dann die aus verschiedenen Universen gewonnenen Kontextinformationen in einer integrierten Form darstellt. Um diese Ansätze zu unterstützen sowie bei sämtlichen involvierten Anwendungen und Kontextmodellteilen ein eindeutiges Verständnis über die beteiligten Realweltobjekte zu erreichen, werden die Informationen in dieser Arbeit basierend auf einer expliziten *Ontologie* definiert, auf welche im folgenden genauer eingegangen wird.

5.2.3 Ontologien

Unter Ontologien wird – wie bereits in Kap. 4.5.6 („Ontologien“, S. 117) ausführlicher geschildert – konform mit GRUBER ein repräsentatives Vokabular zur strukturierten Beschreibung eines Ausschnittes der Welt verstanden, mittels dessen Aussagen und Anfragen hinsichtlich dieser Domäne formuliert werden können, die dann von allen beteiligten Instanzen konsistent verstanden werden (vgl. [Grub93b, Schw01]). Das i. d. R. situationsspezifische Bedeutungs- und Problemlösungswissen wird in dieser Arbeit entsprechend auf der Ebene einzelner Situationen in Form von später noch detaillierter vorgestellten *Szenen*beschreibungen dargestellt.

Ontologien in *Context Maps* modellieren die verwendeten Realweltobjekttypen sowie deren Attribute mittels polyhierarchischer Vererbungshierarchien, wobei eine *specialization inheritance* verwendet wird, d. h. Subklassen enthalten präzisere Informationen als deren Superklasse (vgl. [Atk⁺95, S. 7 f]). Andersherum gesehen stellen Superklassen ungenauere, dafür aber allgemeingültigere Beschreibungen für anwendungsspezifische Informationen dar. Auf die gleiche Weise können auch die zwischen den Realweltobjekttypen bestehenden Zusammenhänge modelliert werden. Dadurch wird es möglich, auch unbekannte Informationen durch Auswertung ihrer Vererbungshierarchie, d. h. Superklassen zumindest ansatzweise einordnen und interpretieren zu können, was eine wesentliche Voraussetzung für deren anwendungsübergreifende Verwendung darstellt.

5.2.4 Szenen

Während Ontologien die auftretenden Objektarten lediglich definieren, modellieren Szenen orthogonal dazu die Beziehung der Objektarten in anwendungsübergreifenden oder -spezifischen Zusammenhängen. Dabei wird unter einer *Situation* eine typische, relativ abgegrenzte Konstellation von konkreten Realweltobjekten verstanden, die durch eine kontextsensitive Anwendung methodisch zu unterstützen ist. *Szenen* modellieren Situationen in einer formalen, wiederverwendbaren Weise. Sie beschreiben, welche Typen von Realweltobjektmodellen in dieser Szene welche *Rolle* spielen können und was sich daraus für spezielle Eigenschaften, Anforderungen und Zielstellungen für die einzelnen Realweltobjekte aufgrund ihrer Rollen ergeben. Der Begriff *Rolle* ist dabei im dramaturgischen Sinne zu interpretieren, d. h. eine Rolle definiert die abstrakten Eigenschaften eines in der betreffenden Szene auftauchenden Charakters. Dabei werden nur die Eigenschaften dieser Rolle festgelegt, nicht jedoch, welches konkrete Realweltobjekt diese Rolle in einer konkreten Situation spielen wird.

Eine Szene besteht folglich aus diversen miteinander assoziierten Rollen. Zu jeder Rolle wird angegeben, von welchem Realweltobjektmodelltyp die Realweltobjektmodelle sein müssen, welche diese

Rolle einnehmen dürfen. Ferner können einer Rolle diverse ggf. für diese Szene relevante Attributwerte zugeordnet werden, welche von den diese Szene spielenden Realweltobjektmodellen erfüllt sein müssen.

Beispiel 15 *In der Szene MENSCHEN FAHREN IN EINEM AUTO steuert ein die Rolle FAHRER spielender MENSCH ein die Rolle AUTO spielendes KFZ zu diversen, die Rollen FAHRTZIEL oder ZWISCHENSTATION spielenden ORTEN, wobei sich weitere, die Rolle PASSAGIERE spielende MENSCHEN im betreffenden AUTO befinden können.*

Aus der Rolle, die ein Realweltobjekt-Typ in einer Szene spielen kann, leiten sich diverse Aufgaben, Handlungsmöglichkeiten und Zielstellungen für das diese Rolle spielende Realweltobjekt ab. Im genannten Beispiel könnte gelten, dass Assistenzsysteme mit dem die Rolle FAHRER einnehmenden Menschen nicht visuell interagieren dürfen, weil dieser permanent auf die Straße blicken muß, was für die die Rolle *Passagier* einnehmenden Personen jedoch nicht gilt.

Indem konkrete *Situationen* im Kontext, also aus konkreten Realweltobjekten bestehende Konstellationen in einem praktischen Anwendungsfall als Instanz einer bestimmten *Szene* erkannt und interpretiert werden, kann das durch die Szene repräsentierte Welt- und Problemlösungswissen auf die aktuelle Situation angewendet werden. Da mit einer Rolle sowohl Aufgaben als auch Zugriffsrechte definiert werden, ist das Rollenmodell auch im Rahmen von Sicherheitsstrategien bedeutsam, indem nur Anwendungen mit einer bestimmten Rolle der Zugriff auf die privaten Datenbestände des Nutzers oder das Recht zur Durchführung gewisser Aktionen in dessen Privatsphäre gewährt wird. Da eine Szene auch Transformationsoperationen für die darin befindlichen Komponenten enthalten kann, ist auch die Veränderung bzw. Anpassung eines Kontextmodells an diese Szene möglich, etwa durch das Hinzufügen oder Entfernen diverser Charaktere.

5.2.5 Situationen

Unter *Situationen* werden im Gegensatz zu Szenen konkret in der Praxis auftretende Konstellationen von Realweltobjekten verstanden, d. h. Situationen stellen gewissermaßen Instanzen von Szenen dar, wobei konkrete Realweltobjekte eine in der Szene definierte Rolle übernehmen. Eine Situation wird dann als Instanz einer Szene angenommen, wenn die Konstellationen von Realweltobjekten im konkreten Realweltmodell mit den Beschreibungen in der Szene in Übereinstimmung gebracht werden können. Beispiel 16 stellt eine mögliche Situation dar, welche der in Beispiel 15 formulierten Szene entspricht.

Beispiel 16 *Die Person JOHN DOE ist der gegenwärtige FAHRER seines AUTOS, welches ein Chrysler Voyager mit dem Kennzeichen "DD-KM 2000" ist. Die Person JEFF MALONE fährt als PASSAGIER mit.*

Entsprechend den in der Szene beschriebenen Rahmenbedingungen könnte der Passagier *Jeff Malone* auf ein im Fahrzeug installiertes Assistenzsystem mittels einer auf einem Bildschirm dargestellten GUI zugreifen, während mit dem Fahrer *John Doe* automatisch sprachbasiert interagiert würde. Weitere Beispiele und Darstellungen, wie Szenen und Situationen modelliert werden, finden sich in Kap. 6.2 („Funktionale Validierung“, S. 175).

In dieser Arbeit soll das Situations- und Szenenmodell lediglich zur modularen Strukturierung von komplexen Kontextinformationen verwendet werden. Weitergehende Aspekte, etwa die Modellierung

der dynamischen Eigenschaften von Szenen und Situationen, die formale Definition der den einzelnen Rollen zugeordneten Handlungsparameter sowie der möglichen Abfolge von Situationen, d. h. Modellierungen, welche Situationen unter welchen Umständen auf andere Situationen folgen können, werden nicht verfolgt, sondern in den Ausblick verschoben. Weiterführende Arbeiten finden sich unter dem Schlagwort *Situation-Awareness* in Kap. 4.1.5 („Situationsmodelle“, S. 101) sowie u. a. in den Arbeiten von MEISSEN (vgl. [Mei⁺04]).

5.3 Wiederverwendungsmodell

Einen der zentralen Punkte dieser Arbeit stellt die *Wiederverwendung* von Kontextinformationen dar, d. h. die einer Anwendung bekannten Informationen sollen durch Eingliederung in das anwendungsübergreifende Kontextmodell auch anderen Anwendungen bekannt gemacht und dadurch von diesen wiederverwendet werden.

5.3.1 Persistente Realweltobjektmodelle

Zu jedem Realweltobjekt soll ein persistentes Realweltobjektmodell analog zu einem *Avatar* erschaffen werden, welches sämtliche dieses Realweltobjekt beschreibenden Kontextinformationen in einer gesammelten Form enthält und in verschiedenen Situationen durch verschiedene Anwendungen verwendet wird, wodurch sämtlichen Anwendungen sämtliche Informationen über dieses Realweltobjekt zur Verfügung stehen.

Diese persistenten Realweltobjektmodelle werden von der betreffenden Person selbst oder einem dafür verantwortlichen *Informationsmanager* gepflegt und können auf einem von ihnen kontrollierten Gerät dargestellt und mittels der in Kap. 5.1.4 („Datenverteilungsaspekte“, S. 137) dargestellten *verteilten Datenhaltungsprinzipien* in das restliche Kontextmodell integriert werden, so dass der Eigentümer der Kontextinformationen auch kontrollieren kann, wann diese Informationen durch wen zugegriffen und verarbeitet werden (vgl. [O.V.04f]). Dieses vor allem sicherheitstechnisch motivierte Prinzip, private Daten auch auf einem privaten Gerät vorzuhalten, wurde von MANIATIS entwickelt und in [Man⁺99] beschrieben.

5.3.2 Invariante Situationsaspekte

Das Wissen, wie die einzelnen Informationen innerhalb einer Situation zu interpretieren sind und wie sich die einzelnen Charaktere daraufhin zu verhalten haben, wird in der betreffenden *Szene* modelliert (vgl. Kap. 5.2.4 („Szenen“, S. 142)). Dies kann innerhalb der betreffenden Szene in einer später noch genauer dargestellten algorithmischen Form erfolgen, wobei die Attribute der Charaktere in der Szene wie Variablen verwendet werden können, wobei sie in einem konkreten Anwendungsfall dann mit den entsprechenden Attributwerten der konkreten Realweltobjekte initialisiert werden.

5.4 Elemente von Context Maps

An dieser Stelle wird ein konkretes Metamodell vorgestellt, welches die im ersten Teil dieses Kapitels erarbeiteten theoretischen Grundlagen praxistauglich umsetzt. Um ein umfassendes Verständnis des

		rel. DB (ERM)	OODB	XML	semWeb/RDF	Power (Topic Maps)	Context Cube	Henricksen	Context Toolkit	Context Fabric	Sentient Computing	Context Framework	Gaia	Schmidt	Context Maps
1	Allgemeine Verwendbarkeit	1	1	3	3	3	2	1	1	3	1		3	1	5
2	Multidimensionalität		2	2	2	2	2		2	1	2	1	1	1	2
3	Dynamik		1	1	1	1	2		1	1	2		2	1	3
4	Datenqualität							2		1		2		1	2
5	Datensicherheit									3					1
6	Integration heterogener CI	1	2	2	2	5	4	2	2	5	4	1	2	1	11
7	Endnutzertauglichkeit					1									1
	Gesamt	2	6	8	8	12	10	5	6	14	9	4	8	5	25

Abbildung 68: Anzahl der mit voller Punktzahl (5 Punkte) erfüllten Anforderungen

Modells zu ermöglichen, werden zunächst die Grundlagen, dann die grundlegenden Modellelemente und schließlich die Gesamtarchitektur des Metamodells vorgestellt.

5.4.1 Basis: Topic Maps

Als Basis des in dieser Arbeit entwickelten Integrations- und Darstellungsmodells für Kontextinformationen werden *Topic Maps* verwendet (vgl. Kap. 4.5.5 („Topic Maps“, S. 116)). Diese Entscheidung beruht vor allem auf der Tatsache, dass Topic Maps als Verfahren zur strukturellen Integration verteilter und heterogener Webdokumente konzipiert wurden und daher sehr viele Dinge in diesem Zusammenhang bereits angemessen und systematisch handhaben können, was sich u. a. darin widerspiegelt, dass sie hinsichtlich der meisten Anforderungen mit voller Punktzahl bewertet wurden, sowohl insgesamt als auch im besonders wichtigen Anforderungskomplex 6 („Integration heterogener Kontextinformationen“). Dies wird in Abb. 68 visualisiert, in welcher Felder mit dem Wert 0 für eine bessere Übersichtlichkeit leer gelassen wurden. Die einzige vergleichbar bewertete Arbeit stellt *Context Fabric* dar, das jedoch aufgrund seiner für diese Arbeit überzogenen Zielsetzung nicht verwendet werden kann und damit aus den möglichen Alternativen herausfällt (vgl. 4.6.4 („Context Fabric“, S. 122)).

Funktional lösen Topic Maps durch die Integration von Informationen aus heterogenen Kontextquellen mittels einer expliziten Beschreibungsebene das *Integrationsproblem* auf eine sehr geeignete Weise, allerdings nicht das in diesem Zusammenhang ebenfalls auftretende *Verteilungsproblem*, da Kontextverbraucher die heterogenen Kontextquellen bei Nutzung von Topic Maps nach wie vor direkt nach Informationsinhalten befragen müssten.

Die trotzdem wie aus Abb. 57 auf Seite 132 ersichtlich nur mittelmäßige Gesamtbewertung von Topic Maps wird vor allem durch einige teils grundlegende, teils auch nur durch die zu allgemeine, nicht auf diese Arbeit bezogene Ausrichtung von Topic Maps verursachte Lücken im Modell verursacht, die dann jeweils eine ganze Palette von Folgefehlern nach sich ziehen. Dazu zählen

- die fehlende Darstellung von Variableninhalten im Datenmodell. Die Topics auf der Metaebene dienen nur zur Repräsentation der einzelnen Datensätze, wobei deren Inhalte nach wie vor in den Kontextquellen verbleiben.

- die fehlende Integration von Logik
- eine fehlende Standardisierung zur Darstellung von Datenqualität
- eine fehlende Standardisierung zur Darstellung von Zugriffsrechten
- fehlende Mechanismen zur Darstellung von externen Kontextquellen
- fehlende Möglichkeiten zur Darstellung von verschiedenen Abstraktionsstufen der enthaltenen Informationen

Im Gegensatz zu anderen verwandten Arbeiten, die häufig ein breiteres Spektrum an Anforderungen eher mittelmäßig adressieren und daher eine bessere Gesamtwertung als Topic Maps erhalten, adressieren Topic Maps nur einige der geforderten Modellaspekte, diese aber in einer sehr geeigneten Form.

Die Realisierung eines sämtliche Anforderungen sehr gut erfüllenden Darstellungs- und Integrationsmodells für Kontextinformationen ist aufbauend auf dem Topic-Map-Standards relativ geradlinig, d. h. ohne *Veränderung* bereits existierender Elemente des Standards möglich, da die angesprochenen Problemstellen im Topic-Map-Standard einfach komplett fehlen anstatt wie bei vielen dadurch besser bewerteten Verfahren halbwegs, aber immer noch unzureichend oder gar unpassend implementiert zu sein. Eine Erweiterung des Topic-Map-Standards ist daher im Vergleich zu den anderen verwandten Arbeiten auch nur hinsichtlich weniger, noch fehlender Modellaspekte notwendig und sollte dadurch im Gegensatz zu insgesamt passenderen Modellen einen geringeren Gesamtaufwand verursachen.

Diese Einschätzung untermauert die (relativ triviale) Erkenntnis, dass komplexe Entscheidungen wie die Auswahl einer geeigneten Grundlage zur Gestaltung komplexer Systeme nicht auf einem einzigen, durch ein Bewertungsmodell gefundenen skalaren Wert, sondern auf der teils fachlichen und teils intuitiven Einschätzung eines Experten unter Berücksichtigung der vielschichtigen Randbedingungen und Zusammenhänge basieren sollte, da viele in diesem Zusammenhang relevante Aspekte nicht oder nur unzureichend so weit formalisiert werden können, dass sie sinnvoll in formale Modelle der Realität integriert werden können.

Bei der nun folgenden Vorstellung des in dieser Arbeit entwickelten *Context Map*-Standards werden die wichtigsten Aspekte des *Topic Map*-Standards erwähnt, jedoch wird ein grundlegendes Verständnis dieses Standards vorausgesetzt. Entsprechende Einführungen in den Topic-Map-Standard finden sich u. a. in [PaHu03], der Standard für eine XML Topic Maps (XTM) genannte XML-basierte Repräsentation von *Topic Maps* findet sich in [PeMo01], während dessen auf *HyTime* basierende, von der ISO standardisierte Variante in [Bie⁺99] dokumentiert ist.

Im folgenden werden zunächst die grundlegenden Komponenten von Context Maps vorgestellt. Dieses Wissen ist zum Verständnis des später vorgestellten Schichtenmodells für Context Maps notwendig, welches das im ersten Teil dieses Kapitels entwickelten theoretische Integrations- und Darstellungsmodell für Kontextinformationen realisiert.

5.4.2 Topics

Die aus dem *Topic Map*-Standard entnommenen Topics repräsentieren beliebige Konzepte innerhalb von *Context Maps* (vgl. Kap. 2.2.1 in [PeMo01]). Je nach Art der Map können dies *ontologische Konzepte*,

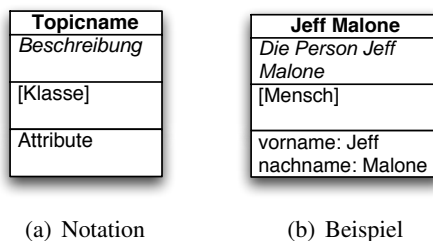


Abbildung 69: Notation für Topics

```

<topic id="jeffMalone">
  <instanceOf><topicRef xlink:href="topicTypes.cm#Mensch" /></instanceOf>
  <baseName>
    <baseNameString>Jeff Malone</baseNameString>
  </baseName>
  <description>Die Person Jeff Malone</description>
  <attribute name="vorname" value="Jeff" />
  <attribute name="nachname" value="Malone" />
</topic>

```

Abbildung 70: XML-Notation für Topics

d. h. Realweltobjekt- oder Assoziationsklassen, *Rollen*, instanziierte Realweltobjekte bzw. Assoziationen, externe Informationen oder Kontextquellen sein.

Abb. 69 zeigt die graphische Notation in allgemeiner Form sowie anhand eines Beispiels. Wie ersichtlich ist, wird der (interne) Name des Topics in fetter Schrift dargestellt. Darunter folgen eine kursiv dargestellte Beschreibung des Topics, in eckigen Klammern die Klasse des Topics sowie in normaler Schrift etwaige Attribute. Dabei müssen abhängig vom jeweiligen Maptyp nicht alle Informationen in den Topics aufgeführt werden.

Abb. 70 zeigt die XML-basierte Notation für Topics. Mittels dem aus dem XTM-Standard entnommenen `instanceOf`-Element, welches eine syntaktische Abkürzung für eine Assoziation vom Typ *class-instance* ist, wird die Klasse dieses Topics durch Referenz auf ein die entsprechende Topic-Klasse repräsentierendes Topic angegeben. Das `description`-Element enthält die textuelle Beschreibung des Topics. Der Name des Topics wird etwas umständlich durch die `baseName`-Struktur angegeben, was im Anschluß explizit erklärt wird.

5.4.2.1 Namensgebung

Die Notwendigkeit zur flexiblen Identifikation aller in *Context Maps* enthaltenen Konzepte wurde durch Anforderung 6k („Synonyme Benennung von Topics“) beschrieben. Zur Identifikation von Realweltobjekten werden in der Praxis durchaus verschiedene Namen verwendet, welche in entsprechender Weise im Realweltmodell dargestellt werden müssen.

Daher wird das bereits im *Topic Map*-Standard verwendete synonyme Identifikationskonzept für Topics ohne weitere Veränderung übernommen. Dabei wird modellintern eine eindeutige *ID* zur Adressie-

```

<topic id="jeffMalone">
  ...
  <baseName>
    <baseNameString>Jeff Malone</baseNameString>
    <variant>
      <parameters>
        <topicRef xlink:href="#nickName" />
      </parameters>
      <variantName>
        <resourceData>Jeffboy</resourceData>
      </variantName>
    </variant>
  </baseName>
  ...
</topic>

```

Abbildung 71: XML-Notation für Namen von Konzepten

ung dieses Konzeptes verwendet, ein *Base Name* zur Präsentation des Konzeptes für den Anwender bzw. als dessen Standardname sowie eine Reihe von *Variant Names* oder kurz *Variants* zur Angabe synonymen Namen für das Konzept definiert. Diese Variants sind optional und werden meistens durch zusätzliche Parameter beschrieben, welche die Bedingungen, unter denen der Variant anstelle des Base Names zu verwenden ist, beschreiben (vgl. Kap. 2.2.2 in [PeMo01]).

Da die graphische Notation vor allem zur intuitiven Visualisierung der Struktur dienen soll, was durch den teilweise recht umständlichen XML-Code nicht immer erreicht werden kann, stellt sie nicht sämtliche Detail Elemente des Context Map Standards zugunsten einer besseren Übersichtlichkeit dar. Dazu gehört die Angabe der synonymen Namen, d. h. ein Topic wird in der graphischen Notation immer mittels seines *Base Names* identifiziert. Werkzeugen steht es natürlich frei, bei Bedarf weitere Informationen über Topics anzuzeigen. Abb. 71 zeigt anhand eines Beispiels die XML-Notation aller genannten Identifikationsmerkmale. Der Variant *Jeffboy* wird hier durch den Parameter *#nickName* als *Spitzname* der Person dargestellt.

5.4.2.2 Identifikation

Unter dem aus dem Topic-Map-Standard weitgehend unverändert übernommenen *Subject Indicator* wird eine weltweit eindeutige ID für ein Topic, welche i. d. R. in Form eines vom Internet bekannten *Uniform Resource Identifier* (URI) dargestellt wird, verstanden (vgl. [Ber⁺98]). Wenn dieser Bezeichner veröffentlicht, d. h. in einer eindeutig beschriebenen, kommentierten und damit wiederverwendbaren Form auch anderen Autoren zugänglich gemacht wird, um auf das betreffende Realweltobjekt damit auch in fremden Maps referenzieren zu können, spricht man von einem *Published Subject Indicator* (PSI) (vgl. Kap. 2.3 in [PeMo01]).

Im Gegensatz zum Topic-Map-Standard, bei dem die als Subject Identifier verwendete URI nur als Identifikator verwendet wird und nicht zwangsläufig auf irgend etwas in diesem Zusammenhang sinnvolles zeigen muß, muß sie bei Context Maps auf das *Topic* in einer *Context Map* zeigen, welches das

```

<topic>
  ...
  <subjectIdentity>
    <subjectIndicatorRef
      xlink:href="http://www.jeffmalone.com/profile.ctm#jeffMalone" />
    </subjectIdentity>
  ...
</topic>

```

Abbildung 72: XML-Notation für *Subject Indicators*

betreffende Realweltobjekt repräsentiert und weitere Informationen zu ihm in Form von Attributen bereitstellt.

Die Notation für (Published) Subject Indicators ist konform mit dem Topic-Map-Standard und wird in Abb. 72 anhand der Person *Jeff Malone* dargestellt.

Ein *Subject Indicator* kann dabei auf drei verschiedene Arten angegeben werden, die in Tab. 20 dargestellt und beschrieben werden.

5.4.3 Occurrences

Gegenüber dem im *Topic Map*-Standard verwendeten Occurrence-Begriff, bei dem Occurrences externe Informationen referenzieren, die das betreffende Topic in einer beliebigen Form repräsentieren, stellen Occurrences in dieser Arbeit beliebige externe Informationen, d. h. Informationen in externen Datenquellen, die einen gewissen Aspekt des Topics beschreiben, inhaltlich im Sinne des in Kap. 5.1.1 („Zugriffsmodell“, S. 135) beschriebenen Zugriffsmodells dar. Im Gegensatz dazu repräsentieren die im Anschluß vorgestellten *Attribute* konkrete Attribute von Realweltobjekten, deren Inhalt u. U. aus diversen miteinander fusionierten Occurrences zusammengesetzt wird.

Zur Adressierung der externen Informationen werden die bereits im Internet zur global eindeutigen Identifikation von Ressourcen verwendeten *Uniform Resource Identifiers* (URI) verwendet (vgl. [Ber⁺98]). Während im originalen *Topic Map*-Standard lediglich Dokumente oder deren Teile referenziert werden müssen, was mittels der *Uniform Resource Locators* (URL) realisiert werden kann, müssen in *Topic Maps* auch einzelne Datenfelder in anderen Quellenarten, etwa relationalen Datenban-

Definitionsart	Bedeutung	Bsp
subjectIndicatorRef	das referenzierte Objekt <i>beschreibt</i> die gesuchte Identität	elektronische Repräsentation eines physikalischen Realweltobjektes durch ein Topic
resourceRef	das referenzierte Objekt <i>ist</i> die gesuchte Identität	ein nicht weiter zu beschreibendes elektronisches Objekt wird direkt referenziert
topicRef	das referenzierte Topic <i>hat</i> die gesuchte Identität	ein anderes Topic mit der gewünschten Identität

Tabelle 20: Definitionsarten für Subject Identity

Name	Typ	Beschreibung
http	Hypertext Transfer Protocol	Zugriff auf Teile von strukturierten, meist auf HTML, XML oder dem Context-Map-Format basierenden Dokumenten
sql	Structured Query Language	Zugriff auf Inhalte in einer relationalen Datenbank
appleAddressbook	Apple Address book	Zugriff auf Informationen in einer Adressbuchdatei auf einem Macintosh-Rechner
outlookAddressbook	MS Outlook Address book	Zugriff auf Informationen im Adressbuch von MS Outlook

Tabelle 21: Schemata von im *Context Map*-Standard verwendeten URI

ken, referenziert werden. Dazu sind weitere Adressierungsinformationen notwendig, die innerhalb der Occurrence dargestellt werden müssen. Dazu wird die Syntax der URI entsprechend erweitert.

Neben den im Internet üblichen Zugriffsverfahren, wobei hier vor allem das *Hypertext Transfer Protocol* (HTTP) und das *File Transfer Protocol* (FTP) zu nennen sind, werden für weitere Quellenarten bzw. deren Zugriffsprotokolle neue *URI-Schemata* erzeugt (vgl. Kap. 3 in [Ber⁺98]). Tab. 21 stellt einige Schemata vor.

Neben der IP-Adresse der Quelle und den gegebenenfalls zu verwendenden Logindaten wird der Ort des gesuchten Datensatzes innerhalb der Quelle in Form eines URI-Parameters angegeben. Es ergibt sich die folgende Syntax für URI in Context Maps: `[Quellentyp]://[Logindaten zur Quelle]@[Adresse der Quelle]?[Ort des Datensatzes in der Quelle]`. Anhang C auf Seite LXXI zeigt einige Beispiele für Occurrences, die Datensätze in typischen Kontextquellen referenzieren.

Um die Informationen über die verwendete Quelle zur Adressierung mehrerer Datensätze zu verwenden, können die quellenbezogenen Teile der URI in einer separaten *Quellenrepräsentation* vorgehalten und mittels des Namens der betreffenden Quelle in eine konkrete Ressourcenreferenz eingefügt werden. Entsprechend wird eine Quelle durch die URI `[Quellentyp]://[Logindaten zur Quelle]@[Adresse der Quelle]` und eine Occurrence durch die Referenz `[Quellenname]?[Ort des Datensatzes in der Quelle]` angegeben, wobei der Quellenname zur Zugriffszeit durch die URI der Quelle zu substituieren ist. Weitere Einzelheiten über Quellenrepräsentationen werden bei der Besprechung von *Beschaffungsmaps* in Kap. 5.4.4.1 auf Seite 152 gegeben.

Occurrences sollen nicht direkt verwendet, sondern nur innerhalb von im Anschluß vorgestellten Attributen eingesetzt werden, um externe Informationen zu adressieren, welche dann in geeigneter Weise verarbeitet und als Attributwerte des betreffenden Topics dargestellt werden. Zu jeder Quellenart, d. h. zu jedem durch ein URI-Schema definierten Zugriffsprotokoll wird von der *Context Map*-Engine gemäß dem in Kap. 5.1.1 auf S. 135 beschriebenen Zugriffsmodell ein eigenes Zugriffsmodul bereitgestellt, welches URI mit dem passenden Schema interpretieren und die dadurch adressierten Informationen zugreifen und für die Engine bereitstellen kann. Die Notation von *Occurrences* in XML ist in Abb. 73 auf der gegenüberliegenden Seite dargestellt.


```

<topic id="jeffMalone">
  ...
  <attribute>
    ...
    <occurrence id="jeffMalone_email_tud">
      <instanceOf>
        <topicRef xlink:href="#email" />
      </instanceOf>
      <resourceRef
        xlink:href="mysql://superuser:pwd@db.tu-dresden.de/?
          accessOperation=doQuery&dbName=staff&
          query=SELECT email FROM personen WHERE user_id=123" />
      </resourceRef>
    </occurrence>
    ...
  </attribute>
  ...
</topic>

```

Abbildung 73: XML-Notation für Occurrences

5.4.4 Attribute

Um die Inhalte der diversen, u. U. synonymen und widersprüchlichen *Occurrences* eines Topics in einer angemessen verarbeiteten Weise darstellen zu können, wird das aus anderen Modellen bekannte Konzept von *Attributen* verwendet, welche in analoger Weise Topics zugeordnet werden und entsprechende Objektattribute inhaltlich darstellen. Attribute enthalten die in Tab. 22 dargestellten Elemente, welche im folgenden eingehender beschrieben werden. Als *Datentyp* von Werten können beliebige atomare Datentypen wie Zahlen, Zeichenketten oder bei komplexeren Attributwerten ein XML-Fragment verwendet werden.

In der graphischen Notation werden Attribute innerhalb der Notation von Topics dargestellt (vgl. Abb. 69 auf Seite 147). Die Notation in XML ist wieder anhand eines Beispiels in Abb. 74 auf der nächsten Seite dargestellt, wobei die Notation der einzelnen Attributelemente erst dann vorgestellt wird, wenn diese Elemente auch vorgestellt werden.

Attributelement	Beschreibung
Name	Name des Attributes
Value	augenblicklicher Wert des Attributes
Occurrences	externe Kontextvariablen, welche den Attributwert enthalten
ProcureMap	Beschreibung des Beschaffungsvorganges des Attributwertes aus den Occurrences
History	Speicherort für vergangene Werte des Attributes
Quality	Beschreibung der Datenqualität des Attributwertes
Security	Regelung des Zugriffs und der Verarbeitung des Attributes
Dynamics	Beschreibung der zu erwartenden Änderungs- und Nutzungshäufigkeit des Attributes.

Tabelle 22: Attributelemente

```

<topic>
  ...
  <attribute name="vorname">
    <value> <resourceData>Jeff</resourceData> </value>
    <procureMap>...</procureMap>
    <history>...</history>
    <quality>...</quality>
    <security>...</security>
    <dynamics>...</dynamics>
  </attribute>
  ...
</topic>

```

Abbildung 74: XML-Notation für *Subject Indicators*

```

<attribute>
  ...
  <value> <resourceRef xlink:href="#unknown" /> </value>
  ...
</attribute>

```

Abbildung 75: XML-Notation für eine unbekannte Kontextvariable

Zusätzlich wird durch den *Context Map*-Standard der Messwert *Unbekannt* festgelegt, der dann verwendet wird, wenn keine Messung möglich ist und in jedem Fall von einer kontextsensitiven Anwendung verarbeitet werden muß. Mit diesem Wert wird ein Kontextmodell initialisiert.

5.4.4.1 Beschaffungsmap

Da die Inhalte von Attributen u. U. aus mehreren (synonymen) Kontextvariablen aus verschiedenen Kontextquellen bestimmt werden, können sie nicht direkt referenziert werden, sondern müssen über einen expliziten Beschaffungsprozess importiert und dabei ggf. miteinander fusioniert werden. Dazu wird innerhalb von jedem Attribut ein Beschaffungsmodul für dessen Attributwert dargestellt, welches ebenfalls als Map realisiert ist und daher *Beschaffungsmap* (engl. *procure map*) genannt wird.

Eine solche Map modelliert die Beschaffung des Inhalts eines Attributes, indem sie sämtliche Kontextvariablen sowie die mit diesen Daten auszuführenden Operationen darstellt. Als Kontextvariablen kommen in diesem Zusammenhang in erster Linie die *Occurrences* dieses Attributes, aber auch Attributwerte von anderen Topics in anderen Context Maps, vor allem aus dem persistenten Avatar des betreffenden Realweltobjektes, in Frage. Operationen stellen vor allem *Transformationen* der Repräsentation einer Kontextvariablen, *Interpretationen* von Kontextvariableninhalten sowie *Fusionen* von verschiedenen Kontextvariableninhalten dar.

Beschaffungsmaps ermöglichen die Trennung der *Informationsbeschaffung* von der *Informationsdarstellung*. Während auf der Realweltmodellebene Informationen über die Realweltobjekte und deren Zusammenhänge dargestellt werden, modellieren Beschaffungsmaps die Beschaffung der einzelnen Attributwerte. Deshalb können sie als *orthogonal* zu den Realweltrepräsentationen interpretiert werden können, was in Abb. 76 auf der gegenüberliegenden Seite dargestellt ist. Indem als Kontextquellen an-

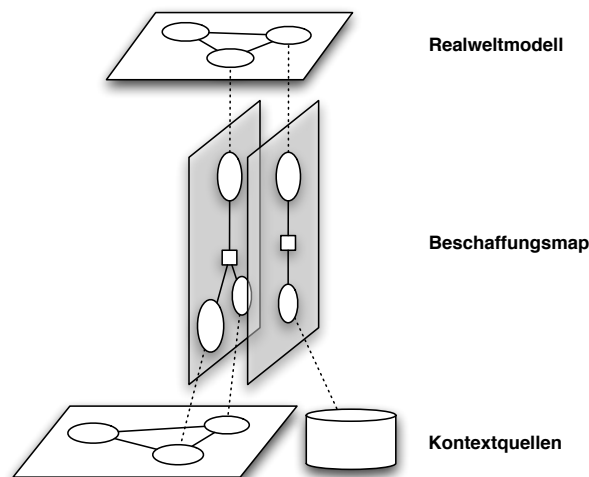


Abbildung 76: Orthogonaler Zusammenhang zwischen Realweltmodellen und Beschaffungsmaps

dere Context Maps verwendet werden, kann mittels Beschaffungsmaps außerdem eine ggf. stufenweise Verdichtung bzw. Abstrahierung der Informationen aus den als Quellen fungierenden Context Maps erreicht werden.

Jedes Attribut verfügt über genau eine *Beschaffungsmap*, welche den Beschaffungsvorgang nur dieses Attributwertes realisiert. Jede Beschaffungsmap enthält das Topic, welches das zu berechnende Attribut enthält sowie ein oder mehrere weitere Topics, welche Occurrences, d. h. die externen Originaldaten aus den Kontextquellen sowie ggf. diese Quellen selbst repräsentieren. Weitere in diesem Zusammenhang verwendete Topics sind durch Aktive Assoziationen zu bestimmen. Diese Topics sind durch diverse *Assoziationen* miteinander verbunden, wobei eine *Aktive Assoziation*, d. h. eine Assoziation mit einem integrierten Codeblock letztlich das *Ergebnistopic* mit einem Wert befüllt. Auf Aktive Assoziationen wird später noch genauer eingegangen.

Ein Beispiel ist in Abb. 77 dargestellt. Hier wird demonstriert, dass die EMail-Adresse von Jeff Malone aus dem Mitarbeiterverzeichnis der TU-Dresden zu entnehmen ist. Dazu wird dessen Personalnummer aus einem bereits existierenden Attribut in seiner Repräsentation entnommen und in der entsprechenden Occurrence verwendet.

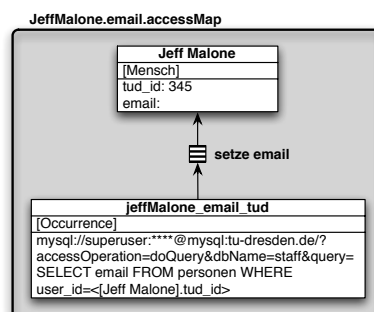


Abbildung 77: Beispiel für eine Beschaffungsmap

```
<topic id="jeffMalone">
  ...
  <attribute id="jeffMalone_email">
    ...
    <history>
      <historicalValue
        timestamp="2000-05-23--14-01"
        value="jeff.malone@mailbox.tu-dresden.de" />
      <historicalValue
        timestamp="1996-10-03--08-00"
        value="jm6@rcs.urz.tu-dresden.de" />
    </history>
    ...
  </attribute>
  ...
</topic>
```

Abbildung 78: XML-Notation für die Historie von Attributen

5.4.4.2 Historie

Die Darstellung vergangener Inhalte von Kontextvariablen, d. h. Attributen von Realweltobjektrepräsentationen ist wie bereits in Anf. 3c („Darstellung der Historie von Kontextvariablen“) dargestellt notwendig für eine ganze Reihe kontextbezogener Entscheidungsvorgänge, u. a. um durch Extrapolation geschätzter Entwicklungstrends zukünftige Situationen und Kontextvariablenwerte zwecks Verwendung in proaktiven Anwendungen schlußzufolgern.

Aufgrund der besonderen Bedeutung historischer Werte im Zusammenhang mit kontextbezogenen Anwendungen werden diese Informationen in dieser Arbeit als notwendiger Bestandteil des Kontextmodells erachtet und daher innerhalb dessen dargestellt. Die Speicherung der historischen Attributwerte erfolgt zusammen mit einem Zeitstempel in einem *Historie* genannten stapelförmigen Speicherbereich im betreffenden Attribut.

Dem Laufzeitsystem ist es so möglich, entweder den im Attributfeld *value* dargestellten *aktuellen* oder einen im Element *history* enthaltenen *historischen* Messwert als Attributwert anzunehmen und dadurch letztlich das gesamte Kontextmodell oder auch nur Teile davon in beliebige Zeitpunkte zurückversetzen zu können, wobei in allen Zeitpunkten die gesamte operative Funktionalität des Kontextmodells, z. B. das darin enthaltene Problemlösungswissen, bereitsteht. Fernerhin ist durch diese Speicherungsform die übersichtliche Darstellung des Entwicklungsverlaufs eines einzelnen Attributwertes über die Zeit möglich, da diese Informationen in gesammelter Form im Attribut vorliegen. Der Historienpeicher für Attributwerte hat den in Abb. 78 dargestellten Aufbau.

Es obliegt dem Laufzeitsystem und nicht dem Kontextmodell, die Attributhistorienspeicher von Zeit zu Zeit von irrelevanten Werten zu bereinigen, d. h. die gespeicherten historischen Attributwerte in geeigneter Form zu verdichten, zu archivieren und aus dem operativen Modell zu entfernen. Die hierfür anzuwendenden Verfahren sollen jedoch nicht Bestandteil des in dieser Arbeit entwickelten Darstellungsmodells für Kontextinformationen sein, da sie eher von den zur Verfügung stehenden Laufzeitressourcen als von modellinternen Parametern abhängig sind.

Attribut	Qualitätsdeskriptor	
	Qualitätsparameter	Qualitätsmetrik
Ort des Nutzers	Genauigkeit	Korrektheit des verwendeten Sensorsystems in Metern pro Sekunde
	Aktualität	Meßzeitpunkt durchschnittliche Lebensdauer einer nutzerbezogenen Ortsinformation
Temperatur im Zimmer	Genauigkeit	Dezikelvin
Aktivität des Nutzers	Sicherheit der Kontextquelle über die Korrektheit ihrer eigenen Angaben	Wahrscheinlichkeitsmaß
	Zuverlässigkeit der Kontextquelle allgemein	Wahrscheinlichkeitsmaß

Tabelle 23: Beispiele für nutzerbezogene Qualitätsdeskriptoren

5.4.4.3 Qualitätsdeskriptoren

Die Notwendigkeit zur Darstellung der Qualität der durch *Context Maps* verarbeiteten Informationen ist aufgrund von deren Kontextbezug essentiell, was durch Anforderung 4 („Datenqualität“) beschrieben wurde.

HENRICKSEN beschreibt ein auf den Arbeiten von WANG basierendes Qualitätsmodell für Kontextinformationen, bei dem die Datenqualität von Attributwerten durch verschiedene attributspezifische *Qualitätsdeskriptoren* modelliert wird (vgl. [Wan⁺95], [Hen⁺02, S. 177 ff]). Dabei wird jeder Qualitätsdeskriptor durch eine Reihe von jeweils eine Qualitätsdimension repräsentierenden *Qualitätsparametern* beschrieben, welche durch attribut- und qualitätsspezifische *Qualitätsmetriken* repräsentiert werden. Ein Beispiel dazu ist in Tab. 23 gegeben. Dieser Ansatz wird ohne weitere Veränderungen in dieser Arbeit verwendet. Die XML-Notation wird in Abb. 79 auf der nächsten Seite anhand eines Beispiels dargestellt. Die graphische Notation entspricht der normalen Notation von Attributen, die jedoch in einer anderen Farbe erfolgen kann, um Qualitätsdeskriptoren von normalen Attributen zu unterscheiden.

5.4.4.4 Sicherheitsdeskriptoren

Der Schutz der mittels kontextsensitiver Verfahren erhobenen personenbezogenen Informationen stellt eine kritische Notwendigkeit für deren Einsatz dar, was durch Anforderung 5 („Schutz der Privatsphäre“) beschrieben wurde. Dort wurde die Notwendigkeit, datenschutzbezogene Informationen innerhalb eines praxistauglichen Modells zur Zugriffskontrolle integrierter kontextbezogener Informationen darzustellen detailliert motiviert.

Context Maps sehen daher eine mit den Arbeiten von HONG konforme Beschreibungen von Zugriffsrechten auf deren interne Informationen vor (vgl. [Hon⁺03, S. 13]). Diese Zugriffsrechte sind von einem diese Maps verarbeitendem Laufzeitsystem zu berücksichtigen. Um sicherzustellen, dass dies auch der Fall ist, sollten nur quelloffene bzw. entsprechend zertifizierte Lösungen verwendet werden, worauf an

dieser Stelle mit Verweis auf Kap. 6.4.2 („Schutz der Privatsphäre“, S. 199) jedoch nicht weiter eingegangen werden soll. Zugriffsrechte können mit einer variablen Granularität auf ganze Maps, Scopes, einzelne Topics oder Attribute vergeben werden. Sie stellen dar, wer der Eigentümer von Informationen ist, wer auf welche Weise auf welche Informationen zugreifen darf bzw. welche Operationen auf welchen Informationen erlaubt sind.

Als *Betroffener* einer Information gilt dabei laut Bundesdatenschutzgesetz – welches an dieser Stelle als geeignete Richtlinie zur Bewertung datenschutztechnischer Belange fungieren soll – diejenige Person, deren „*persönliche oder sachliche Verhältnisse*“ durch diese Information beschrieben werden (vgl. §3 Abs. 1 Bundesdatenschutzgesetz). Als *Eigentümer* von Informationen kann ferner nur der *Betroffene* oder eine von ihm beauftragte Instanz, welche eine Person, Firma oder Anwendung sein kann, auftreten.

Der Zugriff ist daher zunächst nur dem Eigentümer von Informationen oder von ihm beauftragten Instanzen erlaubt. Diese zunächst sehr restriktiven und damit praxisuntauglichen Zugriffsrechte können vom Eigentümer bei Bedarf erweitert werden. Zur Auswahl stehen dabei drei Sicherheitsaspekte, die unabhängig voneinander jeweils automatisch oder erst nach Rückfrage beim Eigentümer gewährt oder verweigert werden können. Dabei kann bei *Zugriffen durch externe Instanzen* die weitere Verwendung der herausgegebenen Informationen prinzipiell nicht mehr kontrolliert werden, was im Gegensatz dazu bei *modellinternen* Operationen noch möglich ist. Daher werden erstere nicht weiter unterschieden, während bei letzteren wiederum *Kopiervorgänge* von *Transformations-* bzw. Fusionsoperationen der betreffende Information in andere Formen unterschieden werden, da bei letzteren eine Veränderung der Detailgenauigkeit der Informationen stattfinden und damit ein veränderter Sicherheitskontext gelten kann.

Entsprechend werden die in Tab. 24 auf der gegenüberliegenden Seite dargestellten *Datenschutzdeskriptoren* für Elemente von Context Maps definiert, wobei als Wertebereich wie bereits erwähnt die automatische Gewährung bzw. Verweigerung des Zugriffs durch bestimmte Instanzen sowie ein einmaliger Zugriff erst nach Rücksprache beim Eigentümer der Informationen (bzw. einer von ihm autorisierten Instanz) gewählt werden kann. Die letztgenannte Idee, die Zugriffsrechte auf private Informationen von einem expliziten *Datenschutzassistenten* im Auftrag des Nutzers zu verwalten, welcher fortgeschrittenere Datenschutzkonzepte basierend auf der Analyse weiterer Kontextbedingungen realisiert, birgt eine

```

<attribute>
  ...
  <quality>
    <qualityDescriptor>
      <qualityParameter> <resourceRef xlink:href="#accuracy" /> </qualityParameter>
      <qualityMetrics>
        <unit> <resourceRef xlink:href="#meter" /> </unit>
        <value> <resourceData>10<resourceData> </value>
      </qualityMetrics>
    </qualityDescriptor>
  </quality>
  ...
</attribute>

```

Abbildung 79: XML-Notation für Qualitätsdeskriptoren von Attributen

Regelname	Beschreibung
accessible by (instance id)	Zugriff durch Instanz erlauben
copyable	Kopieren in andere Maps erlauben
fusable	Vereinigung mit anderen Informationen erlauben

Tabelle 24: Arten von Datenschutzdeskriptoren

Menge interessantes Forschungspotenzial, worauf an dieser Stelle jedoch mit Verweis auf den Ausblick nicht weiter eingegangen werden soll.

5.4.4.5 Dynamikdeskriptoren

Zu unterscheiden sind Zugriffe auf die externen Kontextquellen, welche den jeweils aktuellsten Wert besorgen von der Verwendung der im Kontextmodell zwischengespeicherten Kontextvariablenwerte, die zwar veraltet, dafür jedoch schneller und ggf. ohne Verwendung einer teuren Netzwerkverbindung bzw. langsamen Kontextquelle zugegriffen werden können. Entsprechend ist ein möglichst den aktuellen Umständen angemessener Kompromiss zwischen einem möglichst aktuellen Modellinhalt durch häufiges Abfragen der Kontextquellen und einer möglichst performanten Arbeitsweise des Kontextmodells durch eine möglichst häufige Verwendung der modellintern zwischengespeicherten Kontextvariablenwerte zu wählen. Diese Entscheidungen werden durch das im folgenden dargestellte *Dynamikmodell* realisiert.

Neben der Modellierung von *Aktualisierungsabhängigkeiten*, d. h. der Darstellung, welche Informationen bei Veränderung welcher anderen Information zu aktualisieren sind, gilt es, das *zeitliche Zugriffsverhalten* auf die externen Kontextquellen zu modellieren. Der erste Dynamikaspekt wird durch die noch vorzustellenden *Aktiven Assoziationen* und der zweite durch sogenannte *Dynamikdeskriptoren* modelliert. Diese stellen vor allem zwei Dynamikaspekte dar. Die mindestens einzuhaltende Pause zwischen zwei Zugriffen auf die betreffende Kontextquelle wird durch den Dynamikdeskriptor *MinTimeBetweenUpdates* angegeben. Mehrere Anfragen auf die betreffende Kontextvariable innerhalb dieser Zeitspanne sind durch den modellintern zwischengespeicherten Wert zu beantworten. Um das Kontextmodell auch ohne explizite Zugriffe von Kontextkonsumenten aktuell zu halten, was etwa für kontextbezogene Push-Dienste, die interessierte Anwendungen über Veränderungen im Kontext informieren notwendig ist, kann mittels des Deskriptors *MaxTimeBetweenUpdates* in bestimmten Intervallen auch ohne Konsumentenzugriffe eine Aktualisierung des Kontextmodells veranlasst werden. Die XML-basierte Notation für Dynamikdeskriptoren wird in Abb. 80 auf der nächsten Seite dargestellt.

5.4.5 Assoziationen

Assoziationen stellen Zusammenhänge beliebiger Art zwischen zwei oder mehreren Topics dar. Zusätzlich zum dem bereits vom Topic-Map-Standard bereitgestellten *normalen* Assoziationstyp definieren Context Maps mit *Abstrakten Assoziationen* sowie *Aktiven Assoziationen* zwei zusätzliche Assoziationsarten, die im folgenden näher vorgestellt werden.

```

<attribute>
  ...
  <ynamics>
    <minTimeBetweenUpdates>
      <unit> <resourceRef xlink:href="#second" /> </unit>
      <value> <resourceData>5</resourceData> </value>
    </minTimeBetweenUpdates>
    <maxTimeBetweenUpdates>
      <unit> <resourceRef xlink:href="#hour" /> </unit>
      <value> <resourceData>24</resourceData> </value>
    </maxTimeBetweenUpdates>
  </ynamics>
  ...
</attribute>

```

Abbildung 80: XML-Notation für Qualitätsdeskriptoren von Attributen

5.4.5.1 Normale Assoziationen

Normale Assoziationen entsprechen weitestgehend den aus dem *Topic Map*-Standard bekannten Assoziationen, mit der Ausnahme, dass sie ontologisch definiert sein *müssen*. Um verschiedene Arten von Assoziationen auch semantisch zu unterscheiden, werden Assoziationen deshalb genau wie Topics in einer hierarchischen Klassenhierarchie typisiert.

Die durch diese Assoziation verbundenen Topics werden *Member* der Assoziation genannt. Jedes *Member* spielt eine gewisse *Memberrolle* in einer Assoziation, wobei eine Memberrolle nur von Topics mit bestimmten Typen eingenommen werden kann (vgl. Kap. 2.2.4 in [PeMo01]). Diese Memberrollen beschreiben im Gegensatz zu den in Kap. 5.5.2 („Szenenebene“, S. 165) beschriebenen *Rollen* nur die Rolle des jeweiligen Members in der jeweiligen Assoziation und nicht die Rolle des betreffenden Topics im Weltmodell. Abb. 81 zeigt die graphische Notation für normale Assoziationen. Hierbei wird der Name der Assoziation fett und die Memberrollen kursiv dargestellt.

Abb. 82 auf der gegenüberliegenden Seite zeigt die XML-basierte Notation für normale Assoziationen anhand des in Abb. 81 dargestellten Beispiels.

5.4.5.2 Abstrakte Assoziationen

Abstrakte Assoziationen wurden aus einer ganzen Reihe von Gründen realisiert.

- *Scopes* können die Darstellung der Informationsmenge nur themenbezogen eingrenzen. Mit ihnen ist es jedoch nicht möglich, eine Informationsbasis oder auch nur einzelne Teile davon auf unterschiedlichen Abstraktionsgraden darzustellen. Verschiedene Abstraktionsgrade sind nicht nur zur Handha-

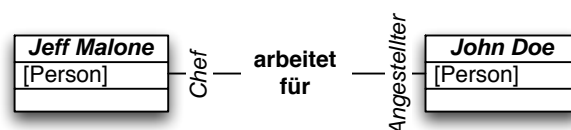


Abbildung 81: Graphische Notation für normale Assoziationen


```
<association id="assoc7465">
  <instanceOf><subjectIndicatorRef xlink:href="#arbeitet_fuer"></instanceOf>
  <member>
    <roleSpec><subjectIndicatorRef xlink:href="#Chef"></roleSpec>
    <topicRef xlink:href="#jeffMalone">
  </member>
  <member>
    <roleSpec><subjectIndicatorRef xlink:href="#Angestellter"></roleSpec>
    <topicRef xlink:href="#johnDoe">
  </member>
</association>
```

Abbildung 82: XML-Notation für eine normale Assoziation

bung der Komplexität notwendig, sondern auch, um Teile von Maps in einer vereinfachten Form zu präsentieren, wobei diverse im Augenblick irrelevante Details verborgen werden sollen, ohne dass die grundlegenden Zusammenhänge in der Map verloren gehen.

- Normale Assoziationen können nur elementare, unmittelbare Zusammenhänge zwischen zwei Topics darstellen. Mittelbare, übergreifende Zusammenhänge, die nicht direkt, sondern nur über mehrere Zwischenschritte hinweg gelten, können mit den Mitteln des Topic-Map-Standards semantisch nicht korrekt modelliert werden, da sie bei einer Darstellung als normale Assoziationen nicht von elementaren Assoziationen unterschieden werden können.
- Ebenfalls stellen *Topic Maps* keine wirklich eleganten Methoden zur wiederverwendbaren Darstellung komplexer Zusammenhänge bereit.

Die geforderte Funktionalität, Teile der Informationsbasis mittels verschieden abstrakter Teilmodelle darzustellen wird durch *Abstrakte Assoziationen* ermöglicht. Dies sind Assoziationen mit zwei alternativen, unterschiedlich abstrakten Sichtweisen auf den von ihnen repräsentierten Zusammenhang. Die *high-level*-Sichtweise zeigt nur die betreffende Assoziation als direkte Verbindung zwischen den indirekt miteinander verbundenen Topics (vgl. Abb. 83(a) auf der nächsten Seite), während in der *low-level*-Sichtweise die einzelnen Zwischenschritte dieser eigentlich indirekten Verbindung eingeblendet werden (vgl. 83(b) auf der nächsten Seite). Diese beiden Darstellungsarten einer *Abstrakten Assoziation* werden auch als *auf-* bzw. *zugeklappte* Darstellung bezeichnet, wobei die *low-level*-Darstellung als *aufgeklappte* und die *high-level*-Darstellung als *zugeklappte* Sichtweise bezeichnet wird.

Abb. 83 auf der nächsten Seite zeigt beide Versionen einer später in Kap. 6.2.4 („Kontextsensitives Navigationssystem“, S. 186) verwendeten Abstrakten Assoziation. Wie ersichtlich ist, wird die *high-level*-Darstellung durch einen kleinen Kreis gekennzeichnet, welcher die von dieser Assoziation repräsentierte, jetzt aber nicht sichtbare *low-level*-Map darstellt. Die *high-level*-Darstellung stellt dar, welche Verabredungen eine Person hat, wobei die *low-level*-Darstellung zeigt, dass diese durch Analyse der E-Mails, Termine und Notizen der betreffenden Person bestimmt werden.

Abb. 84 auf der nächsten Seite zeigt die selbe Assoziation als XML-Code. Zu beachten ist, dass es sich hierbei um die *Definition* der Assoziation und nicht um eine *Instanz* davon handelt. Des Weiteren ist die *low-level*-Ansicht in einer eigenen Map dargestellt, welche mittels des `mapRef`-Tags referenziert wird. Sie könnte jedoch auch direkt innerhalb der `lowLevelMap`-Tags definiert werden.

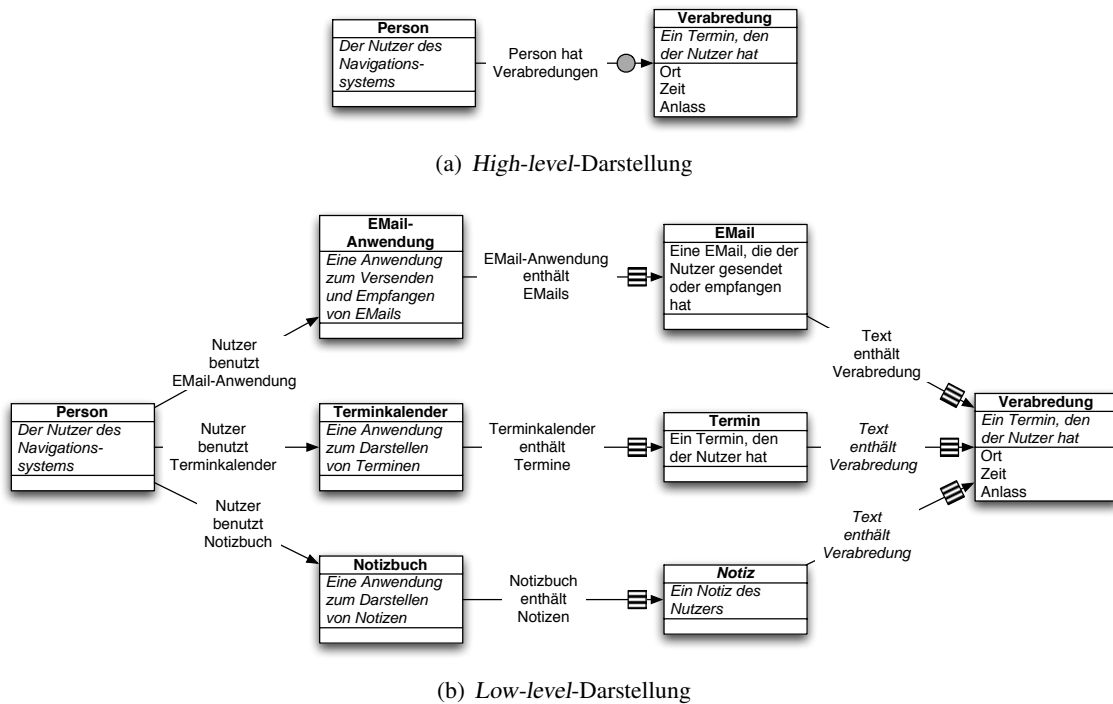


Abbildung 83: Darstellungsarten einer Abstrakten Assoziation

```

<abstractAssociation id="Person_hat_Verabredungen">
  <subclassOf><subjectIndicatorRef xlink:href="#Person_hat_Daten"></subclassOf>
  <member>
    <roleSpec><subjectIndicatorRef xlink:href="memberTypes.cm#Person"></roleSpec>
    <restriction>
      <topicType>
        <subjectIndicatorRef xlink:href="topicTypes.cm#Mensch">
        </topicType>
      </restriction>
    </member>
    <member>
      <roleSpec>
        <subjectIndicatorRef xlink:href="memberTypes.cm#Verabredung">
        </roleSpec>
      <restriction>
        <topicType>
          <subjectIndicatorRef xlink:href="topicTypes.cm#Verabredung">
          </topicType>
        </restriction>
      </member>
    <lowLevelMap>
      <mapRef xlink:href="AbstractAssoc_lowlevelMap_Person_hat_Verabredungen.cm" />
    </lowLevelMap>
  </association>

```

Abbildung 84: XML-Notation für eine Abstrakte Assoziation

Indem sämtliche Abstrakten Assoziationen einer Map zunächst in der einfachen *high-level*-Darstellung präsentiert werden, wird diese Map in einer einfachen und übersichtlichen Form dargestellt. Später sollen bei Bedarf lediglich die gerade aktiven Abstrakten Assoziationen aufgeklappt und damit nur die benötigten Details eingeblendet werden. Dadurch lassen sich auch sehr komplexe und umfangreiche Maps, die eine sehr große Menge von auch sehr detaillierten und umfassenden Informationen enthalten können, in konkreten Anwendungsfällen trotzdem sehr übersichtlich darstellen und verarbeiten.

5.4.5.3 Aktive Assoziationen

Die Integration methodischer Elemente dient zur anwendungsübergreifenden Darstellung und Nutzung informationsbezogener Fusions-, Transformations- und Interpretationsroutinen in der Kontextinformationsbasis und wurde durch Anforderung 6h („Integration von Methodik in das Datenmodell“) motiviert. Weitere Einsatzmöglichkeiten stellen die Berechnung der Validität bzw. Member einer dadurch sehr flexiblen Assoziation zur Laufzeit sowie die Umstrukturierung der Informationsbasis abhängig von der aktuellen Situation ebenfalls zur Laufzeit dar.

Daher wird mit *Aktiven Assoziationen* ein weiterer, diesmal gerichteter, d. h. in einer bestimmten Richtung verlaufender Assoziationstyp definiert, der einen Codeblock enthält, welcher jedesmal ausgeführt wird, wenn diese Assoziation verwendet oder ihr Ausgangsmitglied verändert wird. Dieser Codeblock kann sowohl externe Routinen etwa durch Aufruf einer Web-Service-Schnittstelle aufrufen als auch beliebige Elemente innerhalb der Map verändern. Der Codeblock sollte in einer plattformunabhängigen Form verfasst sein, um möglichst einfach auf andere Geräte replizierbar zu sein. Der in Kap. 6.1 („Realisierbarkeit“, S. 175) vorgestellte *Context Map*-Server verwendet etwa die plattformunabhängige Sprache *Java*. Ebenso gut möglich wären Scriptingsprachen wie etwa *ECMAScript* oder *Python* (vgl. [ECMA99]).

Die graphische Notation für Aktive Assoziationen ist in Abb. 83(b) auf der gegenüberliegenden Seite dargestellt. Wie ersichtlich ist, wird die Richtung einer Aktiven Assoziation durch einen Pfeil dargestellt. Dieser Pfeil reicht im Prinzip schon zur Charakterisierung einer Assoziation als Aktive Assoziation aus, trotzdem sollte zusätzlich das in der betreffenden Abbildung ebenfalls sichtbare gestreifte Viereck angegeben werden, welches die Codezeilen des Codeblockes ikonisiert. Dieses Viereck kann jedoch z. B. aus Platzmangel weggelassen werden. Die XML-basierte Notation einer Aktiven Assoziation ist in Abb. 85 auf der nächsten Seite dargestellt. Der Codeblock wird dabei in einem `codeBlock`-Element vorgehalten, der Rest ist identisch zu einer normalen Assoziation. Das dargestellte Beispiel bestimmt die Nutzer-ID von einem Single-SignOn-Server, indem es dessen Web-Service-Schnittstelle bestimmt, nach der aktuellen Nutzer-ID befragt und diese in das entsprechende Attribut des Nutzer-Topics in der Map schreibt.

5.4.6 Scope

Um die in Anforderung 6 („Einschränkung der Sichtbarkeit von Konzepten“) beschriebenen, bei praxisnahen Szenarien zwangsläufig entstehenden umfangreichen Bestände an Kontextinformationen allgemein besser verwalten und handhaben zu können und insbesondere die dabei mit zunehmender Größe häufiger auftretenden Synonyme zu beherrschen, wird das bereits im Topic-Map-Standard verwendete

```

<activeAssociation id="getUserID">
  ...
  <codeBlock language="javaFragment">
    <![CDATA[
      String ssoServerAddress = getTopic("login server").getAttribute("wsApi");
      WsServer ssoServer = new WsServer(ssoServerAddress);
      String userURI = ssoServer.getUserURI(currentLoginData);
      Topic userTopic = getTopic("user");
      userTopic.setAttribute("userID", userURI);
    ]>
  </method>
  ...
</activeAssociation>

```

Abbildung 85: XML-Notation für eine Aktive Assoziation

Scope-Konzept übernommen und auch auf die in *Context Maps* neuen Elemente wie *Attribute* angewendet.

Ein *Scope* definiert den Geltungsbereich einer in einer *Topic Map* enthaltenen Information. *Scopes* bestehen aus einer Reihe von *Themes*, wobei jedes *Theme* einen einzelnen Begriff darstellt und der *Scope* sich durch eine AND-Verknüpfung aller *Themes* ergibt und damit durchaus sehr spezifisch sein kann. Als Beispiel sei ein aus den beiden *Themes* *Reiseführer* und *Paris* zusammengesetzter *Scope* genannt, der etwa nur diejenigen Personen oder Anwendungen beinhaltet, die als Reiseführer in Paris fungieren können.

Scopes werden neben der themenspezifischen Eingrenzung der *Map* wie bereits erwähnt zur Abgrenzung synonymen *Topics* verwendet. So darf es innerhalb einer *Context Map* zwei *Topics* mit dem selben Namen, auch dem selben *Base Name* geben, solange sie verschiedenen *Scopes* zugeordnet sind. Die durch einen *Scope* dargestellte Information ist auch beim *Merging* von *Topic* und *Context Maps* notwendig, um die Vereinigung synonymen, aber unterschiedlicher *Topics* zu vermeiden (vgl. [PaHu03, S. 20 f]). *Scopes* können jedoch auch zu einzelnen Elementen von *Topics*, etwa deren *Variant Names* zugeordnet werden, wodurch dieser *Variant* dann nur im betreffenden *Scope* gilt. Beispielsweise wird der *Variant* „*Munich*“ dem *Scope* „*en*“ zugeordnet, „*München*“ dagegen dem deutschen *Scope*, wodurch der Name des *Topics* (bzw. der Stadt) in verschiedenen Sprachen definiert und die Darstellung entsprechend sprachenspezifisch angepasst werden kann.

Scope-Informationen werden wie in Abb. 86 auf der gegenüberliegenden Seite dargestellt analog zum *Topic Map*-Standard als Metadaten der betreffenden *Context Map*-Elemente angegeben.

5.5 Schichtenarchitektur

Im folgenden wird genauer auf die Architektur des in dieser Arbeit entwickelten Integrations- und Darstellungsmodells eingegangen, welche das in den Kap. 5.1-5.3 entwickelte theoretische Integrations- und Darstellungsmodell für Kontextinformationen umsetzt. *Context Maps* ergänzen das *3-Schichten-Modell* des *Semantic Web* um eine Ebene zur Modellierung *Szenen* genannter, semantisch und strukturell vorgefertigter Teilmodelle sowie eine Ebene zur wiederverwendbaren Darstellung von externen Kontextquellenarten. Es ergibt sich das in Abb. 87 auf Seite 164 dargestellte *5-Schichten-Modell*. Im Interesse einer

```
...
<variant>
  <scope>
    <subjectIndicatorRef xlink:href="http://www.w3c.org/languages.xtm#en">
  </scope>
  <variantName> <resourceData>Munich</resourceData> </variantName>
</variant>
...
```

Abbildung 86: XML-Notation für den Scope, welcher den englischen Namen der Stadt München beschreibt

möglichst problemspezifischen Terminologie wurden dabei neue, am spezifischen Anwendungsbereich dieser Arbeit orientierte Begrifflichkeiten für die einzelnen Modellebenen gewählt.

Allgemein gesprochen modelliert Schicht 1, bestehend aus Schicht 1a und 1b, das Realweltuniversum und damit die Struktur und Bedeutung des betreffenden Kontextmodells, Schicht 2 stellt das konkrete Kontextmodell dar und Schicht 3, wiederum aus einer Schicht 3a und 3b bestehend, die Beschaffung der diversen Kontextinformationen.

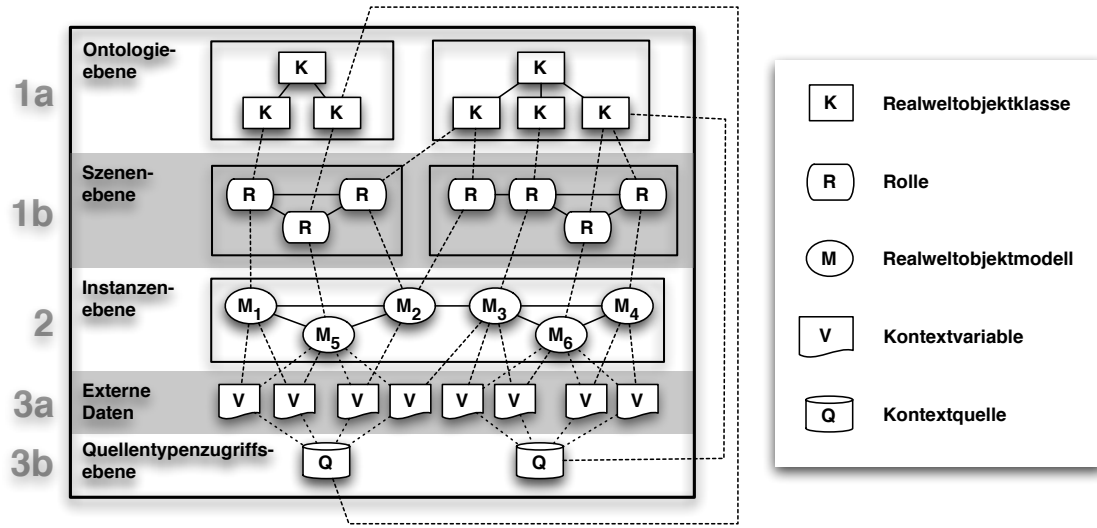
Konkret werden die in externen Kontextquellen befindlichen kontextbezogenen Informationen oder Dokumententeile auf der *Externen Datenebene* (Nr. 3a in Abb. 87) dargestellt und in Attributform zu Realweltobjektmodellen auf der Instanzebene (Nr. 2) zugeordnet. Die Kontextquellen selbst stellen relevanten technischen Kontext der betreffenden Nutzer dar und werden deshalb ebenfalls als Bestandteil des Kontextmodells auf der Instanzebene dargestellt, in Abb. 87 etwa in Form der Objekt M_5 und M_6 . Zum Zugriff auf die betreffenden Kontextquellen werden quellentypspezifische Zugriffsverfahren verwendet, welche mittels expliziter Quellentypmodelle auf der Kontextquellentypenebene (Nr. 3b) beschrieben und konfiguriert werden.

Die Struktur der Kontextmodelle wird auf der Instanzebene durch vorgefertigte, strukturell und semantisch determinierte Teilmodelle, sogenannte *Szenen*, auf der darüberliegenden *Szenenebene* (Nr. 1b) und die semantische Bedeutung der dabei verwendeten Modellelemente schließlich auf der *Ontologieebene* (Nr. 1a) determiniert. Einzelheiten zu den auf den verschiedenen Modellebenen verwendeten Modellbestandteilen werden bei der folgenden Darstellung der einzelnen Modellelemente besprochen.

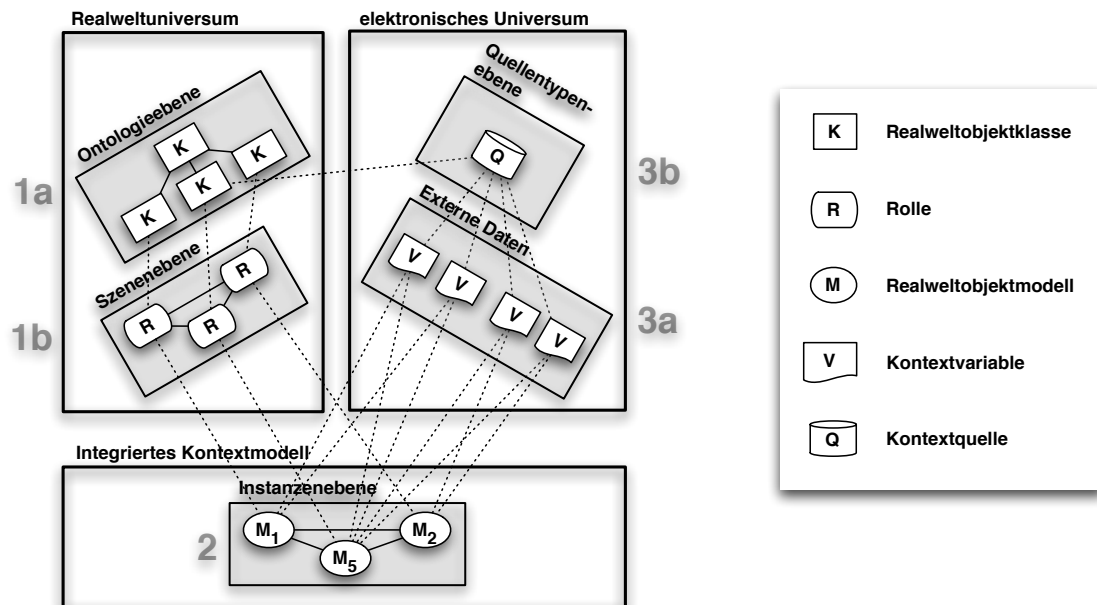
5.5.1 Ontologieebene

Ontologien werden auf der *Ontologieebene* mittels *Ontology-Maps* dargestellt, wobei die zu modellierenden Realweltobjekt- bzw. Assoziationsklassen durch Topics vom Typ *Realweltobjektmodellklasse* repräsentiert werden, die durch Assoziationen vom Typ *superclass-subclass* mit ihren jeweiligen Super- bzw. Subklassenrepräsentationen verbunden sind (vgl. Abb. 88(a) auf Seite 165). Abb. 88(b) stellt ein Beispiel dar, welches angibt, dass ein *Auto* eine Art von *Gerät* ist.

Im Gegensatz zum Topic-Map-Standard, in dem ontologische und inhaltliche Informationen beliebig miteinander vermischt werden können, wird die Vererbungshierarchie bei Context Maps an einer expliziten Stelle modelliert, d. h. ontologische Informationen sind lediglich in *Ontology-Maps* erlaubt, diese sind auf die Angabe von ontologischen Informationen beschränkt, außerdem ist die ontologische Defini-

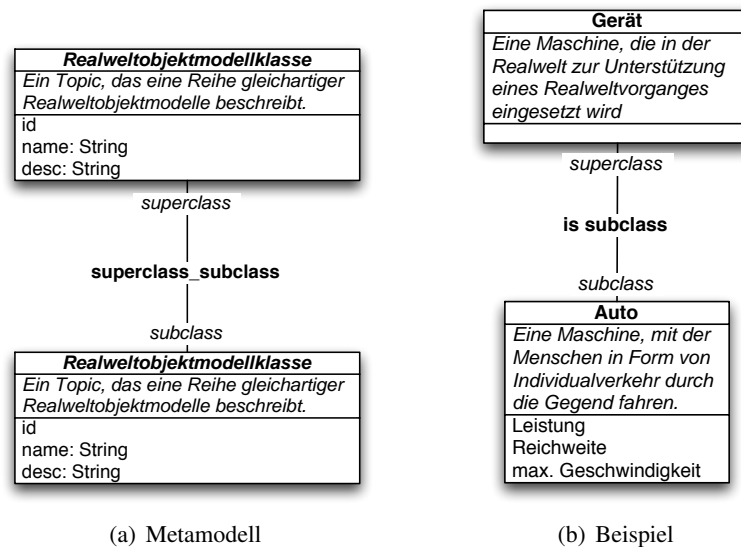


(a)



(b)

Abbildung 87: Schematische Darstellung des 5-Ebenen-Modell von Context Maps

Abbildung 88: Graphische Notation für *Ontology-Maps*

tion aller verwendeten Topics vorgeschrieben. Diese separate Darstellung ontologischer Informationen ist nicht nur strukturierter und übersichtlicher und dadurch besser durch Werkzeuge unterstützbar, sondern kann auch an mehreren Stellen wiederverwendet werden, was für die anwendungsübergreifende Informationsintegration notwendig ist. Diese Regelung wäre auch für Topic Maps sinnvoll und wird daher von vielen Anwendern gefordert, ist aber noch nicht in den Topic-Map-Standard eingeflossen (vgl. [PaHu03, S. 360 ff]).

Um die schrittweise Anpassung der Ontologien an die sich permanent weiterentwickelnde Realwelt zu unterstützen, werden die verschiedenen, durch Erweiterung der Ontologie entstehenden Versionen durch entsprechende Versionsnummern, welche als Attribut der betreffenden *Ontology-Map* dargestellt werden, identifiziert.

Die XML-Notation des in Abb. 88(b) dargestellten Beispiels ist in Abb. 89 auf der nächsten Seite dargestellt. Analog zum durch den XTM-Standard eingeführten `instanceOf`-Tag wird in *Ontologie-Maps* das Tag `subClassOf` als syntaktische Abkürzung für eine Assoziation vom Typ *superclass-subclass* eingeführt. Dieses Element impliziert gleichzeitig, dass das betreffende Topic vom selben Typ wie die Superklasse ist, d. h. das selbe Topic mittels `instanceOf` referenziert.

In Ontologien wird nur die Art der Attribute des Realweltobjekttyps definiert, nicht jedoch deren konkretes Aussehen bzw. deren Beschaffung. Dies erfolgt in der im folgenden beschriebenen Szenenebene, wenn ontologische Konzepte in konkrete Wirkungszusammenhänge eingegliedert werden.

5.5.2 Szenenebene

Während *Ontologien* Realweltobjekte klassifizieren und hinsichtlich ihrer Eigenschaften beschreiben, modellieren *Scene-Maps* die typischen in einem Realweltuniversum auftretenden, *Szenen* genannten Anwendungsfälle, welche aus *Rollen* sowie den zwischen diesen bestehenden Beziehungen und Abläufen bestehen. *Rollen* sind dabei Funktionen, die Realweltobjekttypen in einem Anwendungsfall einnehmen

```

<ontologyMap id="omScanav"
  name="the SCANav ontology"
  version="7.82">

  <topic id="tDevice">
    <name>Gerät</name>
    <desc>Eine Maschine, die ...</desc>
  </topic>

  <topic id="tCar">
    <name>Auto</name>
    <desc>Eine Maschine, mit der ...</desc>
    <subClassOf> <topicRef xlink:href="#tDevice" /> </subClassOf>
    <attribute>
      <name>Reichweite</name>
      <metric>http://www.cm.org/standards/metrics#range</metric>
    </attribute>
    <attribute>
      <name>maxSpeed</name>
      <metric>http://www.cm.org/standards/metrics#speed</metric>
    </attribute>
  </topic>
</ontologyMap>

```

Abbildung 89: XML-Notation für eine *Ontology-Map*

können. Jede Rolle hat dabei individuelle Zielsetzungen, Aufgaben und Rechte. Für Rollen können ebenfalls bestimmte rollenspezifische Attribute definiert werden. Von dieser Möglichkeit wird vor allem in den *Avatare* genannten persistenten Realweltobjektrepräsentationen Gebrauch gemacht. Diese Definitionen können die in den Ontologien gemachten Definitionen ergänzen oder bei gleichem Namen ersetzen.

In *Scene-Maps* werden Rollen durch Topics repräsentiert und durch diverse Assoziationen verbunden, die Beziehungen zwischen den einzelnen Rollen modellieren. Die Attribute der Topics repräsentieren in diesem Fall die Attribute der jeweiligen Rolle. Die graphische Notation ist in Abb. 90(a) auf der gegenüberliegenden Seite dargestellt und wird durch ein Beispiel in Abb. 90(b) veranschaulicht, welches einen Teil der in Beispiel 15 auf Seite 143 dargestellten Situation innerhalb eines fahrenden Autos beschreibt.

Wie aus Abb. 90(b) ersichtlich ist, modellieren Assoziationen in Szenen nicht nur Informationen über den *Aufbau* des Weltmodells, sondern auch über den *Ablauf*, wie dieses Weltmodell zu erschaffen und zu nutzen ist. Als Beispiel stellt die Assoziation *Auto hat Fahrer* nicht nur dar, welche Person als gegenwärtiger Fahrer des betreffenden Autos anzusehen ist, sondern enthält – da es sich um eine Abstrakte Assoziation handelt – auch weitere Informationen darüber, wie die Identität dieses Fahrers bestimmt werden kann, etwa durch Abfrage eines Kontexterfassungssystems im betreffenden KFZ mittels (jetzt nicht sichtbarer) Aktiver Assoziationen in deren low-level-Map. Gleiches gilt für die Aktive Assoziation *Person benutzt Schnittstelle*, welche die jeweils geeigneteste Schnittstelle für die betreffende Person durch Auswertung von deren Situation sowie der verfügbaren Schnittstellentypen im KFZ bestimmt.

Um eine Szene in einem konkreten Anwendungsfall zu nutzen, werden die Realweltobjekte zur Laufzeit mittels des in der betreffenden Szene enthaltenen Weltwissens der für ihre Situation passendsten Rolle zugeordnet, wodurch das im Realweltuniversum dargestellte Problemlösungswissen für die konkrete Realweltsituation nutzbar wird und etwa zur Vervollständigung bzw. Anpassung des Kontextmodells verwendet werden kann.

Einzelne *Scene Maps* modellieren dabei einzelne, separat zu modellierende Teile eines Domänenmodells. Indem dabei Konzepte aus fremden Maps referenziert und somit wiederverwendet werden, kann ein gesamtes Universum modular definiert werden.

5.5.3 Realweltmodellebene

Um die in einem Realweltuniversum definierten Konzepte und Methoden anwenden zu können, werden die Realweltobjekte bzw. persistenten Avatare in der Ontologie des Realweltuniversums dargestellt und ein durch eine Szene definiertes Rollenmodell auf sie angewendet. Dadurch entsteht ein konkretes *Realweltmodell*, welches auf der *Realweltmodellebene* in Form einer Instanz des betreffenden Realweltuniversums repräsentiert wird (vgl. Abb. 91(b) auf der nächsten Seite).

Ein Realweltmodell wird in bereits bekannter Weise durch eine spezielle Art von Maps, die sogenannten *Instanzen-Maps* repräsentiert, in welchen *Topics* die verschiedenen Realweltobjekte und *Assoziationen* die zwischen diesen existierenden Zusammenhänge repräsentieren, wobei die *Attribute* der diversen Topics in diesem Fall die Eigenschaften der konkret dargestellten Realweltobjekte enthalten, welche unter Nutzung von durch ein gleichermaßen mit diesem Realweltmodell assoziierten elektronischen Universum definierten Zugriffsmechanismen aus konkreten Kontextquellen bestimmt werden. Da Universen lediglich abgegrenzte Kontextdomänen beschreiben sollen, können komplexere kontextsensitive Anwendungssysteme durchaus mehrere alternative Universen verwenden, wobei jedes Universum dann lediglich die für eine Domäne typischen Charakteristika modelliert, vorausgesetzt, die diversen Universen verwenden kompatible bzw. durch Universentransformationen entsprechend kompatibel gemachte On-

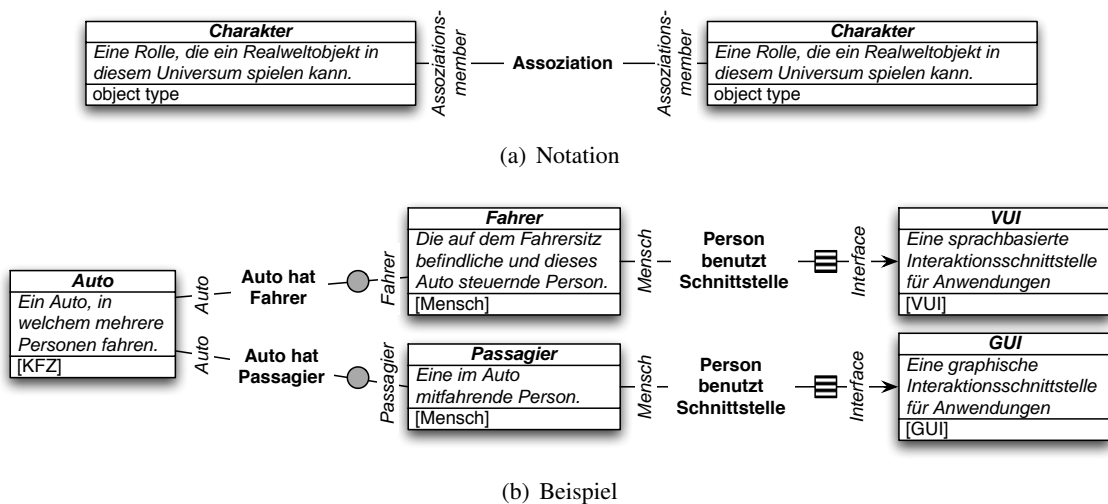


Abbildung 90: Graphische Notation für *Template-Maps*

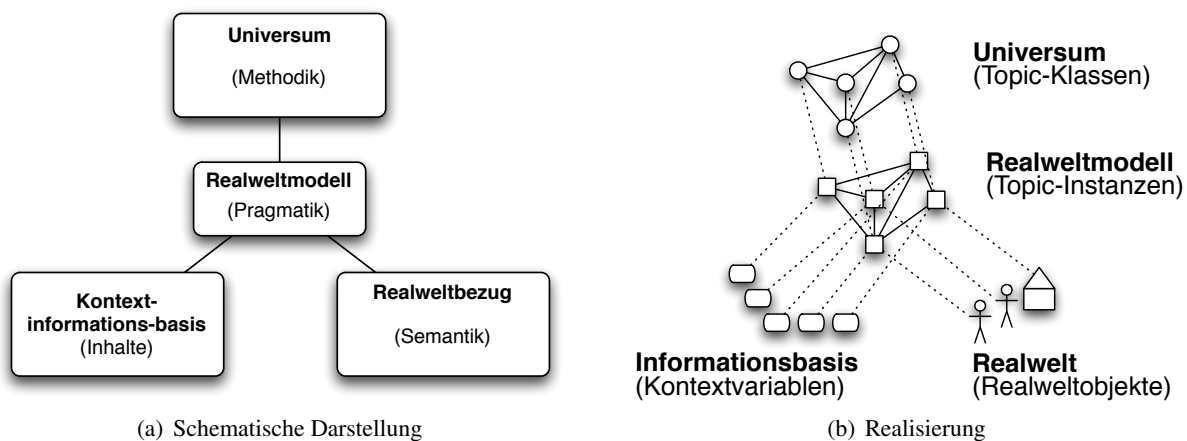


Abbildung 91: Kombination der separat modellierten Modellelemente zu pragmatischen Realweltmodellen

tologien. Die in den Instanzenmaps befindlichen Definitionen etwa hinsichtlich Beschaffungsmaps von Attributen ergänzen bzw. ersetzen die im betreffenden Universum gemachten Angaben. Durch dieses Mittel lässt sich also das durch Universen bereitgestellte generische, wiederverwendbare Strukturmodell durch subjektbezogene Strukturinformationen erweitern.

Während ein Universum sowie die darin enthaltenen Szenen semantische Informationen enthalten, stellen *Situationen* Anwendungen dieses Problemlösungswissens auf einen konkreten Anwendungsfall in der Realwelt dar und haben entsprechend einen pragmatischen Charakter (vgl. Abb. 91(a)).

Die graphische Notation von *Instance-Maps* wird in Abb. 92 durch die Darstellung der in Beispiel 16 auf Seite 143 beschriebenen Situation veranschaulicht.

5.5.4 Externe Datenebene

Das im elektronischen Universum anzusiedelnde Zugriffsmodell ist in analoger Weise in Schicht 2 instanziiert, wird jedoch, um es von den realweltbezogenen Konzepten zu unterscheiden, in Schicht 3 definiert. Auf Schicht 3a (Externe Datenebene) werden die einzelnen externen Datensätze, die sich in diversen externen Datenquellen befinden, in Form der bereits in Kap. 5.4.3 („Occurrences“, S. 149) ausführlich vorgestellten *Occurrences* in einer einheitlichen Form gemäß dem in Kap. 5.1.1 auf S. 135

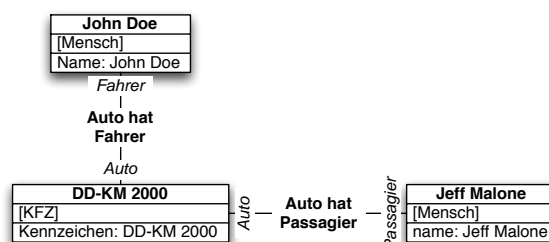


Abbildung 92: Graphische Notation für *Instance-Maps*

dargestellten Zugriffsmodell dargestellt. Die externen Datenquellen selbst werden in Form eines *elektronischen Universums* in einer zu den Realweltuniversen analogen Weise auf Ebene 3b definiert und als Bestandteil des technischen Kontextes auf Ebene 2 (Instanzenebene) dargestellt.

5.5.5 Kontextquellentypenebene

Ebene 3b enthält die *elektronischen Universen*, welche nicht die *Realwelt*, sondern die *elektronische Welt*, d. h. die Kontextquellentypen in den per Netzwerk erreichbaren Rechnersystemen sowie Wege, diese zuzugreifen, beschreiben.

Als *Kontextquelle*, d. h. Objekt, das Kontextvariablen enthält, wird gemäß dem Ansatz dieser Arbeit prinzipiell jedes irgendwie befragbare Realweltobjekt, d. h. im engeren Sinn jede Anwendung und jedes Gerät und im weiteren Sinn sogar auch Menschen verstanden, wobei sich diese Arbeit jedoch eindeutig auf Anwendungen und Geräte, die automatisiert abgefragt werden können, fokussiert, was schon aus dem Namen „elektronisches Universum“ ersichtlich wird. Mit Menschen müsste explizit durch ein geeignetes Endgerät interagiert werden, etwa um Passwörter für Dienste oder Bestätigungen für bestimmte Aktionen einzuholen. Für jedes als Kontextquelle geeignete Realweltobjekt können typische *Zugriffsoperationen* definiert werden, um auf dieses Realweltobjekt im Sinne einer Kontextquelle zuzugreifen und die darin befindlichen Kontextinformationen auslesen zu können. Diese Zugriffsoperationen haben eine nach außen einheitliche Form und kapseln dadurch die spezifischen Details des Quellenzugriffs, d. h. durch die Wahl einer geeigneten Zugriffsoperation können beliebige Kontextvariablen in beliebigen Kontextquellen in einer standardisierten Form zugegriffen werden. Solche Zugriffsmechanismen zur Bereitstellung heterogener Informationen über eine standardisierte Schnittstelle werden bereits seit geraumer Zeit zur Integration heterogener Informationsmengen verwendet, weshalb eine ganze Reihe von Lösungen in diesem Bereich existieren. LEI nennt dieses Konzept *Context Driver*, DEY und Abowd sprechen von *Context Widgets*, MENA von *Wrappern* (vgl. [Lei⁺02, S. 3 f], [Dey⁺01b, S. 19 f], [Men⁺98, S. 5]). In dieser Arbeit werden Zugriffsmodule konform mit MENA als Modul, welches die spezifische Datenorganisation einer externen Kontextquelle kennt und daher den Zugriff auf diese Quelle bei gleichzeitiger Verbergung von deren interner Struktur vor dem zugreifenden Informationssystem ermöglicht, verstanden.

Da die als Kontextquellen infrage kommenden Realweltobjekttypen bereits auf der Ebene 1a in einer geeigneten Weise ontologisch definiert werden, ist dies an dieser Stelle nicht mehr notwendig, sodass durch das Zugriffsmodell nur noch die Zugriffsmöglichkeiten auf die einzelnen, auf der Ebene 1a definierten, Realweltobjekttypen angegeben werden müssen. Dies erfolgt durch *Kontextquellentypenmodelle* auf der Kontextquellentypenebene. Diese geben für einen Kontextquellentyp neben dem in den Occurrences zu verwendenden URI-Schema die darauf anwendbaren Zugriffsoperationen, d. h. deren Name, die benötigten Parameter sowie den Rückgabebetyp an. Abb. 93 auf der nächsten Seite zeigt ein Beispiel für ein Kontextquellentypmodell, die entsprechende DTD findet sich in Anhang B.4. Da Kontextquellentypen keine komplexen Gebilde bilden können, wird auf eine graphische Notation dieses Konzeptes verzichtet.

Diese Zugriffsoperationen können entsprechend in Occurrences verwendet werden und sind auch innerhalb des Codeblockes von Aktiven Assoziationen als Methoden der entsprechenden Kontextquellen-

```

<contextSourceType>
  <realworldObjectType>
    <topicRef xlink:href="#tAppleAddressbook" />
  </realworldObjectType>
  <uriSchema>appleAddressbook</uriSchema>
  <accessOperation>
    <name>getContactIDByName</name>
    <description>
      Liefert die ID des anhand des übergebenen Namens identifizierten Datensatzes
    </description>
    <parameter>
      <name>personName</name>
      <type>String</type>
    </parameter>
    <result>
      <type>String</type>
    </result>
  </accessOperation>
</contextSourceType>

```

Abbildung 93: Beispiel für ein Kontextquellentypmodell

instanzen verfügbar. Die konkrete Realisierung dieser Zugriffsverfahren ist Sache des entsprechenden Laufzeitsystems und an dieser Stelle nicht von Interesse.

5.6 Richtlinien zur Benutzung

An dieser Stelle werden Richtlinien zur Verwendung von *Context Maps* gegeben, die neben einer einheitlichen Verwendung durch alle Anwendungen ein tiefergehendes Verständnis dieses Metamodells liefern sollen.

5.6.1 Strukturelle Veränderung von Universen

Da sich die Realwelt in einem permanenten Veränderungsprozess befindet, müssen auch die betreffenden Kontextmodelle schrittweise aktualisiert werden. Neue Objekt- und Verbindungstypen können auftreten sowie existierende Typen verändert bzw. erweitert werden. Dies betrifft sowohl Ontologien als auch Szenen. Dabei ist darauf zu achten, durch Veränderung von Universen bereits funktionierende Kontextverbraucher nicht in ihrer Funktion zu beeinträchtigen.

Daher dürfen bestehende Universen strukturell nicht *verändert*, sondern höchstens *erweitert* werden, d. h. es dürfen lediglich weitere Modellelemente zum Modell hinzugefügt, jedoch keine existierenden Elemente verändert oder entfernt werden. Um die verschiedenen Versionen eines Universums voneinander unterscheiden zu können, werden diese mit schon aus der kooperativen Softwareentwicklung bekannten Versionsnummern gekennzeichnet. Erweiterungen sollen sich dabei stets in einer *höheren* Versionsnummer des Universums niederschlagen. Bereits existierende Anwendungen können so auch neuere Versionen eines Universums verwenden, wobei sie neue, unbekannte Konzepte zumindest anhand ihrer bekannten Superklasse zumindest grob interpretieren oder auch einfach ignorieren können.

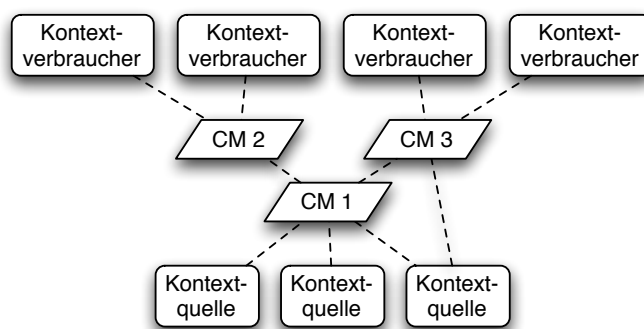


Abbildung 94: Transformation einer *Context Map* in verschiedene Ontologien

Als Beispiel sei das in Kap. 6.2.4 („Kontextsensitives Navigationssystem“, S. 186) vorgestellte kontextsensitive Navigationssystem genannt, welches die E-Mails des Nutzers hinsichtlich möglicher Terminabsprachen analysiert. Neuere Versionen des Universums könnten nun weitere textuelle Kommunikationsformen, etwa Chats, beinhalten. Indem diese durch eine Superklasse *textuelle Kommunikation* gekennzeichnet werden, kann die Navigationsanwendung, die zwar die konkrete Klasse *Chats* nicht, dafür aber die angegebene Superklasse kennt, die Chat-Nachrichten trotzdem zumindest als „textuelle Kommunikation“ interpretieren und deren Inhalte daraufhin in analoger Weise nach Terminabsprachen durchsuchen.

Sind die vorgenommenen Änderungen jedoch derart neuartig, dass eine Weiterführung des alten Universums für neue Anwendungen keinen Sinn mehr macht, d. h. werden strukturelle *Veränderungen* im Universum notwendig, ist ein neues Universum mit einem neuen Namen zu erstellen.

5.6.2 Transformation in andere Ontologien

Um die Informationen in einer Map in einer anderen Ontologie darzustellen, ist es notwendig, die Informationen in dieser Map in einem anderen Universum, d. h. in der durch ein anderes Universum repräsentierten Weltansicht und Begriffswelt zu formulieren. Diese Operation wird *Universentransformation* genannt.

Dabei wird eine neue Map basierend auf dem neuen Universum erstellt und deren Informationen, d. h. Attributwerte ausschließlich aus der alten Map beschafft, d. h. die Transformation der Ontologien wird in den Beschaffungsmaps der Attribute in der neuen Map realisiert. Dieser Zusammenhang wird durch Abb. 76 auf Seite 153 sowie die resultierende Architektur durch Abb. 94 dargestellt, in welcher die Informationen aus der Context Map 1 in die beiden anderen, auf verschiedenen Ontologien basierenden Maps 2 und 3 konvertiert werden. Während Map 2 ausschließlich Informationen aus Map 1 verwendet, greift Map 3 auf eine zusätzliche Kontextquelle zu und stellt deswegen nicht nur eine *Universentransformation*, sondern auch eine *Erweiterung* von Map 1 dar.

An dieser Stelle ist es unerheblich, ob die Universentransformation automatisiert oder manuell erfolgt. Ebenfalls nicht weiter eingegangen soll an dieser Stelle auf die dafür notwendigen interontologischen Informationen, welche angeben, wie die verschiedenen Konzepte in den einzelnen Universen miteinander korrelieren (vgl. [Men⁺98], Kap. 7.2 („Ausblick“, S. 208)).

5.6.3 Abfrage von Kontextinformationen

Ebenfalls festzulegen ist, wie genau das Kontextmodell zu verwenden ist. Zwar stellt auch dieser Aspekt ein implementationsbezogenes Merkmal dar, jedoch soll an dieser Stelle zumindest der grobe Ablauf vorgestellt werden.

Ein Kontextverbraucher stellt dem Kontextdienst das von ihm verwendete Universum zur Verfügung bzw. referenziert das von ihm verwendete Universum mittels einer entsprechenden URI, worauf der Kontextdienst das betreffende Universum importiert. Daraufhin können Kontextverbraucher Anfragen an den Kontextdienst im durch dieses Universums repräsentierten Weltverständnis stellen und bekommen sie entsprechend beantwortet.

Dazu übergibt ein Kontextverbraucher ein mit den ihm bereits bekannten Informationen – etwa über sich selbst – gefülltes und ansonsten leeres Weltmodell an den Kontextdienst, welcher dieses dann basierend auf dem im Universum modellierten, d. h. durch Aktive Assoziationen dargestellten Problemlösungswissen und seinen Möglichkeiten, externe Kontextquellen zuzugreifen, ausfüllt und an den Kontextverbraucher zurückgibt. Nun kann der Kontextverbraucher darin nach den benötigten Informationen recherchieren, indem er ausgehend von z. B. seiner eigenen Repräsentation durch Verfolgen von Assoziationen schrittweise weitere assoziierte Informationen erschließt.

5.7 Anwendungsformen des Modells

Context Maps stellen ein flexibles Mittel zur integrierten Darstellung beliebiger heterogener Kontextinformationen dar und können daher in diverser Hinsicht eingesetzt werden. An dieser Stelle werden die wichtigsten Verwendungsmöglichkeiten für *Context Maps* vorgestellt. Eine weitergehende Beschreibung der aus dem gesamten Konzept erwachsenden betriebswirtschaftlichen Potenziale findet sich in Kap. 3 („Betriebswirtschaftliche Potenziale umfassender Kontextsichtweisen“, S. 49).

5.7.1 Informationsbasis über Realweltobjekte

Der naheliegendste und hauptsächliche Anwendungsbereich für eine auf *Context Maps* basierende Informationsbasis ist die Versorgung kontextsensitiver Anwendungen und ggf. auch Nutzer mit den benötigten Kontextinformationen.

Dabei werden die durch *Context Maps* dargestellten Informationen von kontextbezogenen Anwendungen zugegriffen und für ein kontextsensitives Verhalten verwendet. In diesem Zusammenhang lösen *Context Maps* das dabei auftretende, in Kap. 1.2.2 („Problemstellung“, S. 8) dargestellte Verteilungs- und Integrationsproblem und ermöglichen dadurch den einfachen Zugriff auf eine breite Palette verteilter und heterogener Kontextinformationen.

Speziell durch *Context Maps* können diese Informationen dabei in einer sehr gut verwendbaren, auch für untrainierte Nutzer relativ gut verständlichen Form präsentiert werden. Durch die Integration verteilter und heterogener kontextbezogener Informationen in ein einziges, integriertes, umfassend aussagefähiges Kontextmodell können für die Mehrzahl der Anwendungen sonst unerreichbare Informationen für diese zugreif- und verwendbar gemacht werden, weiterhin können durch die Kombination unterschiedlicher Kontextinformationen weitere kontextbezogene Informationen geschlußfolgert und die

Kontextinformationsbasis dadurch nochmals erweitert werden. Durch die Bereitstellung eines umfassenden Kontextmodells ermöglichen *Context Maps* eine einheitliche, kontextbezogene Funktionalität einer breiten Palette von heterogenen kontextbezogenen Anwendungen.

5.7.2 Situationsmodellierung

Context Maps können jedoch nicht nur zur Angabe von *Kontextinformationen*, sondern auch zur Formulierung von *Kontextsituationen*, d. h. zur Erkennung typischer Situationen im Kontext durch ein kontextsensitives Laufzeitsystem verwendet werden.

Eine Situation wird dabei durch eine *Szene* beschrieben. Die Situation wird dann gegeben angenommen, wenn sämtliche Realweltobjekte den selben Zustand, d. h. die selben Attributinhalt wie die Topics in der Szene haben. Um für die gesuchten Attributinhalt einen Wertekorridor angeben zu können, müssen dabei ggf. zwei Maps verwendet werden, wobei dann eine Map die Minimalwerte und die andere die Maximalwerte angibt.

Vorteilhaft an der Nutzung von *Context Maps* zur Beschreibung von Situationen ist, dass zur Formulierung von Kontexten nicht nur Attributwerte, sondern dank *Aktiver Assoziationen* auch durchaus variable Verbindungen zwischen Realweltobjekten berücksichtigt werden können. Außerdem können Situationen semantisch aussagefähig beschrieben werden, was auch für die daraufhin zu ergreifenden Maßnahmen im Rahmen eines *Situation-Aware Computing* gilt. Dadurch kann das vom *Desktop Computing* bekannte Paradigma des *Event-Driven Computing* in der durch Context Maps ermöglichten ergonomischen, strukturierten und semantisch verankerten Form realisiert werden.

5.7.3 Interaktionsplattform für kontextsensitive Entitäten

Das Prinzip offener Informationsforen basiert auf der Annahme, dass es in einer Umgebung mit gegen Null tendierenden Kommunikations- und Infrastrukturkosten wie dem Internet für den einzelnen kleinen Informationskonsumenten vorteilhafter ist, sein begrenztes Wissen etwa über bestimmte Details einer Problemdomäne in einem offenen Informationsforum zu veröffentlichen, damit diese Informationsbausteine mit anderen, auf gleiche Weise veröffentlichten Informationsbausteinen kombiniert ein großes, umfassendes Informationsgebilde ergeben, welches dann für alle Teilnehmer in vollem Umfang zur Verfügung steht. Entsprechend ist der für jeden Teilnehmer durch den Zugriff auf das gesammelte Wissen entstehende Nutzen wesentlich größer als die von ihm dafür bereitgestellte Informationsmenge. Die Motivation, darin auch selbst Inhalte bereitzustellen und nicht nur „Trittbrett zu fahren“ liegt dabei darin, die Form dieses Informationsgebildes durch entsprechende Beiträge auch hinsichtlich der eigenen Interessengebieten zu gestalten, dadurch weitere Beiträge anderer Nutzer in diesem Gebiet zu verursachen, wodurch dann die erwünschten Synergieeffekte entstehen.

Durch die Kombination der Idee offener Informationsforen mit dem Context-Map-Prinzip, sämtliche Kontextinformationen über bestimmte Kontextobjekte hinsichtlich diverser Problemdomänen in einer flexiblen, strukturierten und semantisch aussagefähigen Form darzustellen, ergibt sich die Vision einer umfassenden *Kommunikationsplattform für realweltbezogene Aspekte* eines Problembereiches, auf der sich durch die Einordnung des Wissens sämtlicher Wissensträger über die betreffende Realweltdomäne ein für alle nutzbares, umfassendes und damit sehr brauchbares Kontextmodell quasi von selbst ergibt.

Dieses Szenario geht von dem heute bislang nur für menschliche Aufgabenträger üblichen Paradigma aus, sich die zur Problemlösung benötigten Informationen zunehmend aus einem öffentlichen Sammelplatz für Informationen zu besorgen, wobei hier allem das World Wide Web zu nennen ist. Es spricht jedoch nichts dagegen, dieses Paradigma auch auf maschinelle Aufgabenträger anzuwenden.

Dabei könnten etwa sämtliche im Auftrag und unter der Kontrolle des Nutzers agierende Anwendungen und Assistenten eine gemeinsame Plattform pflegen, auf der sie sämtliche Informationen über ihren Nutzer zusammentragen. Jeder Assistent verwendet dann keine eigene Infrastruktur zur Erfassung kontextbezogener Informationen mehr, sondern besorgt sich diese auf der Interaktionsplattform und stellt dort seine Erkenntnisse ebenfalls zur Verfügung. Falls er dort nicht ausreichend fündig wird und daher keine für den Nutzer zufriedenstellende Funktionalität erbringen kann, obliegt es dem Nutzer, selbst an der Kommunikation seiner Anwendungen teilzunehmen und die fehlenden Informationen oder Verfahren, diese Informationen zu beschaffen auf dieser Plattform bereitzustellen. Dabei werden von den Anwendungen neben Attributinhalt auch Assoziationen bereitgestellt, um ihr Wissen über bestimmte Zusammenhänge auszudrücken.

Context Maps könnten eine Basis für solche Ansätze darstellen, wobei jedoch weitere Steuerungs- und Koordinationsmechanismen notwendig wären, um ein unkontrolliertes Wuchern der so gewonnenen Informationsmenge zu verhindern.

5.8 Zusammenfassung

In diesem Kapitel wurde ein auf einer fundierten theoretischen Grundlage basierendes Metamodell zur Darstellung für integrierte kontextbezogene Informationen beschrieben. Dieses kombiniert ein *Informationsintegrationsmodell*, bestehend aus einem Zugriffs-, Integrations-, Verbindungs- und Datenverteilungsmodell, mit einem *Kontextmodell*, wiederum bestehend aus einer expliziten Repräsentation der ontologischen Begrifflichkeiten, des Problemlösungswissens sowie einem Situations- bzw. Szenenmodell.

Das Metamodell basiert auf dem Topic-Map-Standard und besteht aus 3 grundlegenden Schichten, welche entsprechend den eben erwähnten theoretischen Modellen die formale Struktur, die Herkunft der zu integrierenden heterogenen Kontextinformationen sowie deren eigentliche Darstellung realisieren. Das Modell beinhaltet neben Verfahren zur synonymen Benennung von Realweltobjektmodellen flexible Mechanismen, heterogene Informationen in einer standardisierten Form zuzugreifen und mit bestehenden Informationen zu integrieren, stellt Verfahren zur Beschreibung der Qualität, Zugriffsrechte sowie der Dynamik von Modellelementen bereit, erlaubt die Definition unterschiedlicher Abstraktionsstufen für Teile der Informationsbasis, die Einbettung aktiver Elemente zwecks Formulierung von Problemlösungs- und Strukturwissen sowie eine Methode zur Abgrenzung diverser überlappender Modellteile hinsichtlich flexibel zu definierender Realweltaspekte.

Neben Richtlinien zur Realisierung des Modells wurden mögliche Verwendungsmöglichkeiten für das Kontextmodell genannt, wobei deren betriebswirtschaftliche Aspekte separat in Kap. 3 („Betriebswirtschaftliche Potenziale umfassender Kontextsichtweisen“, S. 49) behandelt wurden.

6 Validierung

In diesem Kapitel wird eine Validierung des in Kap. 5 entwickelten Verfahrens zur Integration und Darstellung heterogener Kontextinformationen vorgenommen.

Dies erfolgt in drei Schritten. Zuerst wird in Kap. 6.1 die technische Realisierbarkeit der in dieser Arbeit beschriebenen Konzepte demonstriert. Basierend darauf wird in Kap. 6.2 die Funktionalität des Ansatzes anhand der erfolgreichen Modellierung einer funktionierenden Kontextinformationsbasis in einem repräsentativen Anwendungsszenario bewiesen. Anschließend wird das Kontextmetamodell mittels des in Kap. 2.4 („Anforderungsanalyse“, S. 25) dargestellten Bewertungsverfahrens den verwandten Arbeiten mit dem Ziel gegenübergestellt, dessen deutlich bessere Eignung hinsichtlich der durch das Bewertungsverfahren formalisierten Bewertungskriterien darzustellen. Abgeschlossen wird dieses Kapitel durch eine rückblickende Diskussion der am Anfang dieser Arbeit aufgestellten Thesen unter dem Blickwinkel der nun verfügbaren Lösung.

6.1 Realisierbarkeit

Die technische Durchführbarkeit der in *Context Maps* verwendeten Konzepte wurde durch die Entwicklung eines *LINCME* genannten *Context Maps*-Dienstes demonstriert (vgl. [Fehr04]).

Dieser ist in der Lage, *Context Maps* darzustellen und persistent zu speichern, außerdem wurden die in dieser Arbeit beschriebenen *Aktiven Assoziationen* und *Abstrakten Assoziationen* sowie Funktionen zum Mergen von Maps implementiert. Ebenfalls wurden Zugriffsmodule auf E-Mails, Termini, Kontakte und Notizen in MS Outlook implementiert und erfolgreich verwendet.

Dabei wird der Codeblock von *Aktiven Assoziationen* in der Programmiersprache *Java* formuliert. Diese hat den Vorteil, zum einen ohne Modifikation des Quelltextes oder Kompilates plattformübergreifend ausführbar zu sein, wobei entsprechende Laufzeitumgebungen für jede verbreitete Plattform und Geräteklasse verfügbar sind, und zum anderen mittels *Java Reflection* zur Laufzeit kompiliert, instanziiert und ausgeführt werden zu können (vgl. u. a. [FoFo04]). *Abstrakte Assoziationen* werden durch eine separate Map dargestellt, die bei Bedarf in die zu ergänzende Map gemergt wird, wobei die *high-level*-Perspektive der Assoziation entfernt wird. Es werden temporäre und persistente Maps unterschieden, wobei die persistenten Maps in einer relationalen Datenbank gespeichert werden.

Da die Engine nicht als standalone-Anwendung, sondern als eigenes Objekt realisiert ist, kann sie und damit die Fähigkeit, *Context Maps* zu verarbeiten, von jeder java-basierten Anwendung verwendet werden.

6.2 Funktionale Validierung

Als Domäne für die an dieser Stelle zu erzeugenden Kontextmodelle wird ein umfassendes Anwendungsbeispiel in einem fortgeschrittenen kontextbezogenen Szenario verwendet, denn nur in solchen finden sich Kontextquellen in der einen Einsatz von *Context Maps* rechtfertigenden Vielzahl und Heterogenität. Um den in solchen Szenarien möglichen Anwendungskomfort zu demonstrieren, wurden bewusst zahlreiche und heterogene Kontextquellen integriert, von denen sich einige derzeit noch im Entwicklungsstadium befinden, aber technisch in absehbarer Zeit realisierbar sind. Um den durch *Context*

Maps entstehenden funktionalen Mehrwert gegenüber existierenden kontextsensitiven Verfahren darzustellen, wurden weiterhin möglichst viele bereits kontextsensitiv implementierte Anwendungsbereiche in dieses Szenario eingebunden, um den durch existierende Lösungen auf diesem Gebiet entstehenden Nutzen möglichst direkt mit den Mehrwerten von Context Maps vergleichen zu können.

Als Beispielszenario dient eine Reise des Nutzers *Jeff Malone* nach Paris. Angekommen am Flughafen sucht er als erstes eine Autovermietung auf, welche einige der in seinem Reisepass enthaltenen elektronisch lesbaren biometrischen Merkmale nach Rücksprache mit ihm als Autorisierungscode für sein Mietfahrzeug verwendet. Dabei nutzt er auch die optionale Möglichkeit, dem Assistenzsystem im Mietwagen einen Link auf sein Nutzerprofil mitzuteilen, um dessen Dienste zu personalisieren.

Der Mietwagen gewährt die Benutzung des Fahrzeugs folgerichtig erst nach einer erfolgreichen biometrischen Identifikation von Jeff. Da Jeff sich in Paris nicht auskennt, gibt er vor Antritt der Fahrt an einem berührungssensitiven Bildschirm im Fahrzeug das Fahrziel für das Navigationssystem ein. Dank seines verfügbaren Nutzerprofils kann er dabei auf Informationen in seinem Adressbuch oder EMail-Verkehr zugreifen, wobei er die in einer Buchungsbestätigungsemail seines Reisebüros genannte Hoteladresse wählt. Diese Anwendung wird in Kap. 6.2.4 („Kontextsensitives Navigationssystem“, S. 186) genauer vorgestellt.

Sobald das Fahrzeug rollt, wird die Interaktion automatisch auf eine akustische Modalität umgestellt, da der aktuelle Benutzer Jeff gleichzeitig der Fahrer ist. Dabei wird das Stimmprofil von Jeff, welches durch dessen vorherige Interaktionen mit sprachgesteuerten Anwendungen erstellt wurde, aus dessen Nutzerprofil eingelesen, um die Interaktionssprache auf Englisch festzulegen sowie das Spracherkennungssystem auf seine individuelle Aussprache zu eichen. Zunächst greift Jeff mittels der in Kap. 6.2.2 („Messenger-VUI“, S. 179) dargestellten Anwendung auf die während seines Fluges nach Paris neu eingetroffenen E-mails zu. Anschließend führt er einige Telefonate mit Bekannten und Kollegen.

Um Paris kennenzulernen, macht der Nutzer einen Stadtbummel, wobei er von einem in Kap. 6.2.5 („Tourist Guide“, S. 193) detaillierter vorgestellten kontextsensitiven Touristenführer durch Paris navigiert wird. Sein Mittagessen nimmt er in einem beliebten ortstypischen vegetarischen Restaurant in der Nähe ein, welches ihm von der in Kap. 6.2.3 („Restaurant-Finder“, S. 181) vorgestellten Restaurant-Finder-Anwendung gemäß seiner Präferenzen vorgeschlagen wurde.

6.2.1 Nutzerprofil

In sämtlichen Anwendungen wird das *Nutzerprofil* des Anwenders verwendet, das im Prinzip sämtliche bekannten Informationen über den Nutzer enthalten soll, die dazu aus externen Anwendungen zusammengetragen werden. Aus Sicherheitsgründen können diese nutzerbezogenen Informationen von einem eigenen Gerät unter der expliziten Kontrolle des Nutzers verwaltet werden, welches Informationen nur nach einer gründlichen Prüfung etwa der Identität der anfragenden Verbraucher sowie des durch Sicherheitsdeskriptoren beschriebenen Sicherheitsmodells im Kontextmodell herausgibt (vgl. [Man⁺99]).

Zwar stellt ein Nutzerprofil keine wirkliche Situation dar, trotzdem wird es durch ein Realweltuniversum formal beschrieben. Der für das gegebene Szenario relevante Ausschnitt der auf der Szenenebene befindlichen Nutzerprofilbeschreibung ist in Abb. 95 auf der gegenüberliegenden Seite dargestellt.

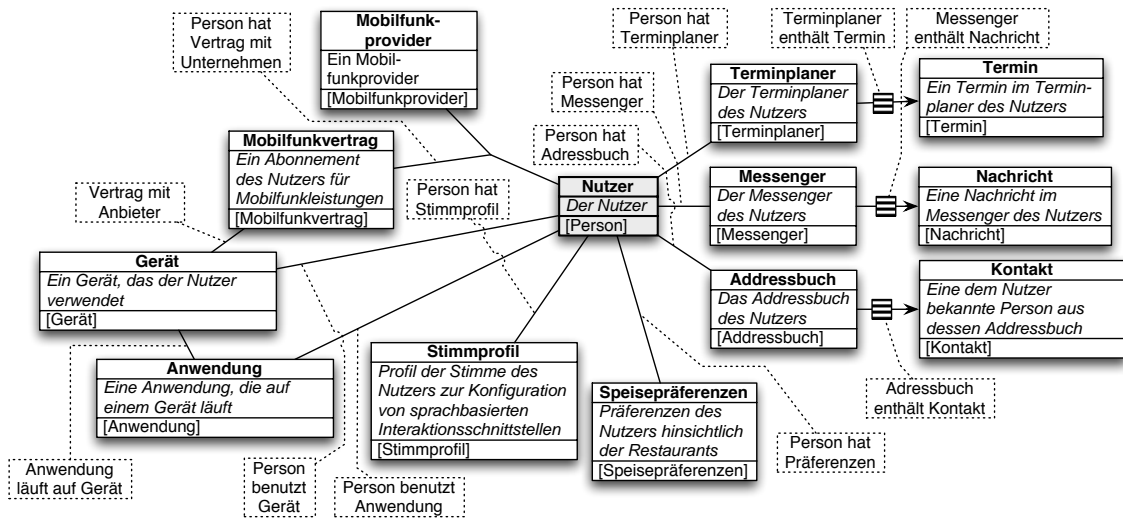


Abbildung 95: Realweltuniversum des Nutzerprofils im Ausgangszustand des Szenarios

Dessen zentrales Element ist das Topic *Nutzer*, welches Jeff repräsentiert und von dem aus auf weitere mit ihm assoziierte Informationen verlinkt wird. Dies sind u. a. seine Termine, EMail und Kontakte, die aus den entsprechenden Anwendungen, d. h. seinem Terminplaner, Messenger und Adressbuch gewonnen werden. Da Jeff Wert auf eine vegetarische Ernährung legt und wegen seines oft straffen Zeitplans auf seinen Reisen nicht lange nach einem entsprechenden Restaurant suchen möchte, hat er entsprechende Speisepräferenzen in seinem Nutzerprofil hinterlegt. Gleiches gilt für die oft von ihm verwendeten sprachgesteuerten Schnittstellen, denen Jeff erlaubt hat, in seinem Nutzerprofil ein Stimmprofil anzulegen und im laufenden Betrieb zu aktualisieren. Auch hat er eine Liste mit den von ihm verwendeten Endgeräten zur Nutzung freigegeben, da er diese teuren Geräte auch möglichst effektiv nutzen möchte, außerdem die Informationen über seinen Mobilfunkprovider sowie -vertrag, um mittels dieser Informationen möglichst preisgünstig in Paris kommunizieren zu können. Sein Ort wird direkt als Attribut seines Topics („Nutzer“) dargestellt, da der gegenwärtige Ort eines Realweltobjektes kein eigenständiges Realweltobjekt ist. Abb. 96(a) zeigt die betreffende Beschaffungsmatrix dieses Attributes. Wie ersichtlich

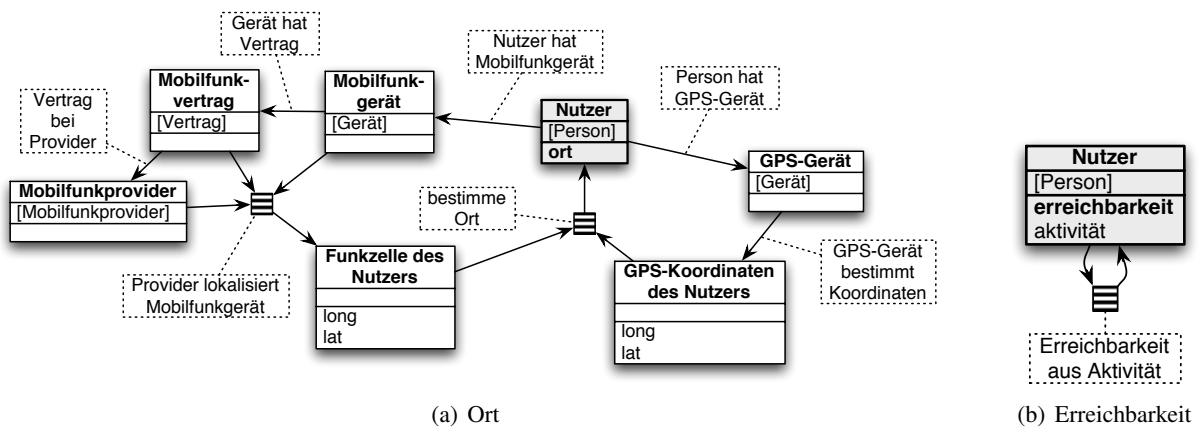


Abbildung 96: Beschaffungsmatrix für Attribute des Topics *Nutzer*

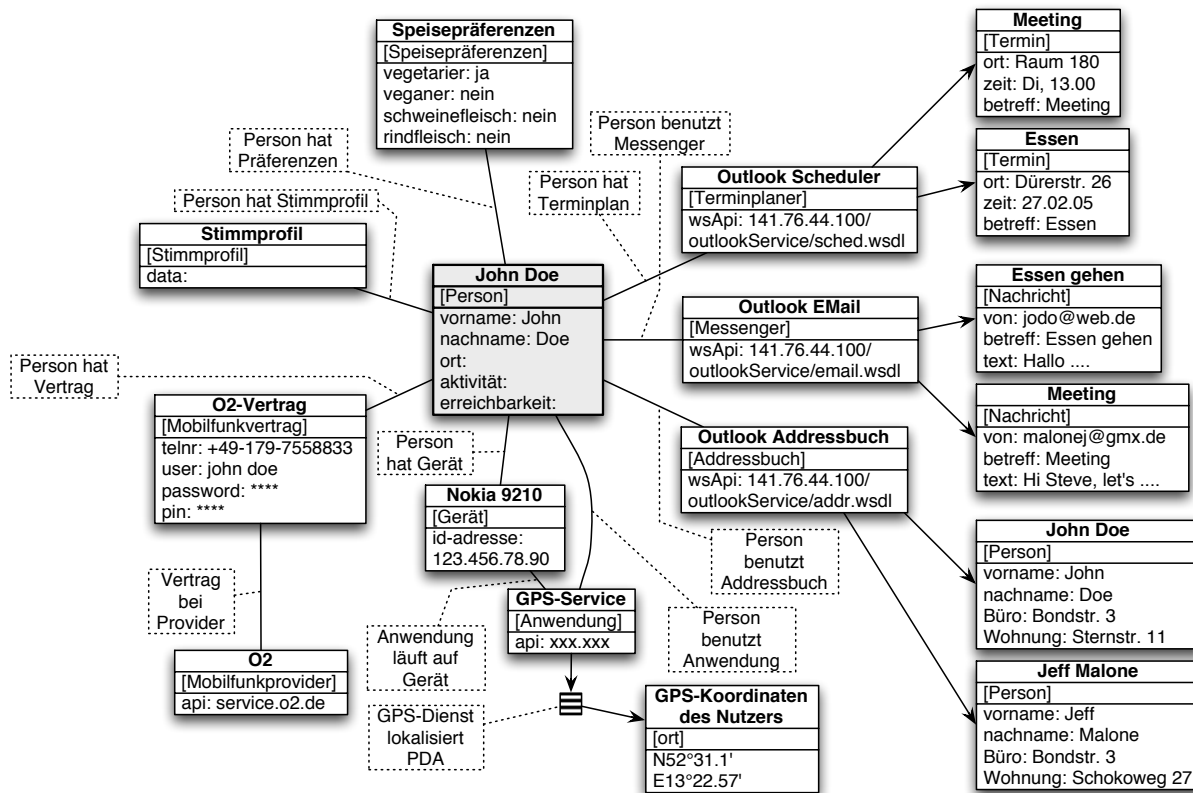


Abbildung 97: Instanz des Nutzerprofils im dargestellten Szenario

ist, kann eine Lokalisierung mittels eines eventuell verfügbaren GPS-Systems auf einem der mobilen Endgeräte, durch ein mobilfunkbasiertes Lokalisierungsverfahren oder durch eine Kombination von beiden Verfahren durchgeführt werden. Die abstrakte Beschreibung dieses Vorganges in der szenischen Beschreibung auf Ebene 1b stellt die dazu notwendigen Schritte dar.

Dabei werden zunächst die Mobilfunkverträge für die gegenwärtig vom Nutzer verwendeten Endgeräte bestimmt, um die entsprechenden Mobilfunkprovider um eine Lokalisierung dieser Endgeräte zu bitten. Parallel dazu werden die ebenfalls gerade vom Nutzer verwendeten GPS-fähigen Geräte herausgesucht und auf diesen ebenfalls eine Lokalisierung veranlasst. Die Ergebnisse beider Operationen werden in der Beschaffungsmatrix in den Topics *Funkzelle des Nutzers* bzw. *GPS-Koordinaten des Nutzers* dargestellt, welche als Eingabeparameter der Aktiven Assoziation *bestimme Ort* dienen, welche den aussagefähigeren Wert ausgewählt und in das gewünschte Zielformat konvertiert.

Im Gegensatz dazu soll die *Aktivität* des Nutzers nicht automatisch bestimmt werden, da diese Kontextvariable zu komplex ist und durch Informationen aus prinzipiell jeder Kontextquelle beschrieben werden könnte. Dieses Attribut soll daher von beliebigen Anwendungen bei entsprechenden Informationen direkt mit einem Inhalt beschrieben werden. Der *Erreichbarkeitsstatus* wird wie aus Abb. 96(b) auf der vorherigen Seite ersichtlich aus der gegenwärtigen Tätigkeit des Nutzers bestimmt, sobald ein solcher Wert verfügbar ist.

Dieses Universum wird im Laufe seiner Verwendung von diversen Anwendungen erweitert, was an der jeweils gegebenen Stelle dargestellt wird. Abb. 97 auf der gegenüberliegenden Seite zeigt das auf diesem

Realweltuniversum basierend in der konkreten Situation am Flughafen in Paris existierende Nutzerprofil von Jeff.

6.2.2 Messenger-VUI

Die erste Anwendung demonstriert vor allem die Nutzung von Informationen aus dem Nutzerprofil. Während der Fahrt mit seinem KFZ verlangt der Fahrer Zugriff auf seine inzwischen neu eingetroffenen E-Mails. Da für den Fahrer während der Fahrt nur sprachliche Interaktion erlaubt ist, damit dieser seinen Blick nicht von der Straße wenden muß, wird eine sprachbasierte Nutzerschnittstelle, ein sogenanntes *Voice User Interface* (VUI) aktiviert. Dieses liest die Betreffzeilen aller ungelesenen E-Mails vor. Von wichtigen E-Mails lässt sich der Fahrer auch den Text vorlesen und löscht oder beantwortet einige E-Mails im Rahmen der bei sprachlicher Interaktion gegebenen Möglichkeiten. Anschließend ruft er über die selbe VUI einige Personen an, wobei er keine Telefonnummern eingeben muß, sondern lediglich den Namen oder Spitznamen der anzurufenden Personen nennt.

Von Interesse ist an dieser Stelle nur das Kontextmodell des VUI der Messenger-Anwendung (*Messenger-VUI*) und nicht deren Aufbau und Funktionsweise. Das Messenger-VUI besorgt sich aus dem Datenbestand des *in-car-Systems*, welches sämtliche Geräte und Anwendungen innerhalb des betreffenden KFZ verwaltet, die Identität des Nutzers sowie den von ihm angegebenen Link auf sein Nutzerprofil. Dies wird in Abb. 98(b) auf der nächsten Seite dargestellt. Die beiden Aktiven Assoziationen bestimmen dabei jeweils mittels des im Auto installierten Bussystems (etwa eines CAN-Bus) zunächst die Adresse der zentralen Steuereinheit im Auto und befragen diese dann nach der biometrisch festgestellten Identität des Nutzers und der Adresse seines Nutzerprofils. Aus diesem, bereits vorgestellten, Nutzerprofil werden dann die Informationen über das E-Mail-Konto, Adressbuch sowie Stimmprofil des Nutzers, welche das Spracherkennungsmodul des Messenger-VUI auf die spezielle Aussprache des Nutzers einstellen, abgefragt. Da das Nutzerprofil ebenfalls als Context Map vorliegt, sind zwei verschiedene Verfahren möglich, um die Informationen in beiden Kontextmodellen miteinander zu integrieren. Zum einen könnten die Informationen analog zu dem in Abb. 98(b) auf der nächsten Seite dargestellten Vorgehen mittels Aktiver Assoziationen aus dem Nutzerprofil herausgesucht und in entsprechende Platzhalter im Kontextmodell des Messenger-VUI integriert werden. Andererseits könnte auch gleich das gesamte Nutzerprofil bzw. dessen freigegebene Teile mittels einer Universentransformation in eine vom Messenger-VUI angegebene Form gebracht und dann durch *Merging* mit dem in Abb. 98(a) auf der nächsten Seite dargestellten Realweltuniversum in dieses integriert werden. Diese Funktionalität würde ebenfalls durch Aktive Assoziationen in Form von Steuerkommandos an das Context-Map-Verarbeitungssystem realisiert werden, worauf an dieser Stelle jedoch nicht näher eingegangen werden soll.

Weitere Informationen sind für das Messenger-VUI nicht erforderlich, d. h. diese Anwendung kann ihre Funktionalität komplett aus den bereits im KFZ sowie Nutzerprofil enthaltenen Kontextinformationen bestreiten und muß keine weiteren Konfigurationselemente oder gar eine eigene kontextbezogene Infrastruktur zur Beschaffung der benötigten Informationen bereitstellen. Das Messenger-VUI ist folglich der *dritten Welle* von kontextsensitiven Anwendungen zuzuschreiben, da es ausschließlich auf bereits in anderen Kontextquellen existierende Informationen zugreift und keine eigene kontextbezogene Infrastruktur benötigt (vgl. Kap. 2.2 („Arbeitsthesen“, S. 21)).

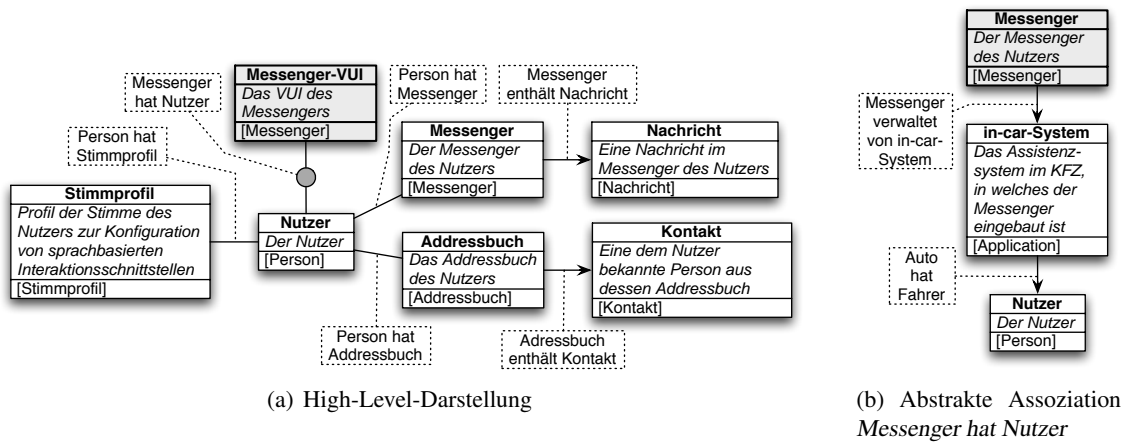


Abbildung 98: Realweltuniversum des Messenger-VUI

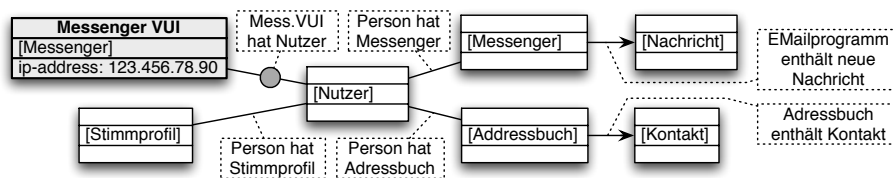


Abbildung 99: Anfrageschablone für die Benutzung des Kontextmodells des Messenger-VUI

Um Informationen aus seinem Kontextmodell zuzugreifen, übergibt das Messenger-VUI lediglich das in Abb. 98(a) dargestellte Realweltuniversum seines Kontextmodells an den *Context Map*-basierten Kontextdienst. Anschließend erzeugt das VUI eine leere Instanz und füllt sie mit den ihm bekannten Informationen, in diesem Fall nur über sich selbst und übergibt sie als Anfrage an den Kontextdienst (vgl. Abb. 99). Dieser bestimmt die passenden Realweltobjekte und Kontextquellen bzw. falls verfügbar deren persistente Avatare, führt die mittels der Aktiven Assoziationen angegebenen Abfragen der Kontextquellen durch, trägt diese so weit es ihm möglich ist in die Modellinstanz ein und gibt sie an den Messenger zurück. Ein Beispiel ist in Abb. 100 auf der gegenüberliegenden Seite dargestellt. Wie ersichtlich ist, gibt es zwei neue Nachrichten für den Nutzer.

Das konkrete Vorgehen des Kontextdienstes beim Bearbeiten der Anfrage wird bei der Besprechung der nächsten Anwendung erklärt.

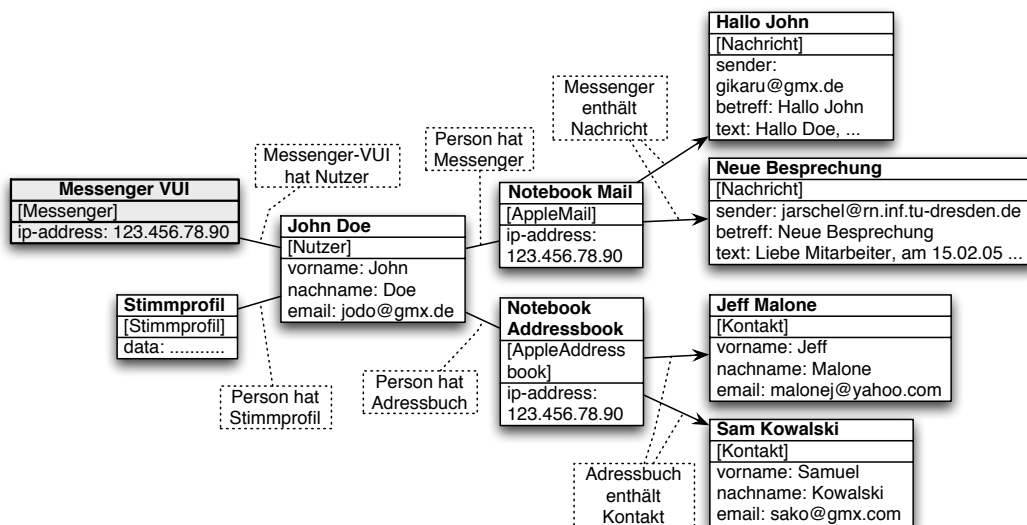


Abbildung 100: Ergebnis

6.2.3 Restaurant-Finder

Anhand des bereits in Beispiel 1 auf S. 7 vorgestellten komfortablen *RestaurantFinders* wird der schrittweise Aufbau einer Instanz eines Kontextmodells im Rahmen der Beantwortung einer Anfrage demonstriert. Die Anwendung zeigt mögliche Restaurants in der Nähe des Nutzers an. Anstatt jedoch einfach nur eine Liste von relativ nichtssagenden Restaurantnamen in der Nähe des Nutzers anzuzeigen, filtert sie diese Liste gemäß dessen Präferenzen und ermöglicht es dem Nutzer dadurch, gezielt ein wirklich *geeignetes* Restaurant auszusuchen, statt einfach nur ein *naheliegendes*. Dazu nutzt die Restaurantfinder-Anwendung die Funktionalität eines bereits existierenden Lokalisierungssystems und erweitert sie um von ihr selbst bereitgestellte kontextbezogene Informationen und Funktionalitäten, weshalb sie der *zweiten Welle* kontextsensitiver Anwendungen zuzurechnen ist (vgl. Kap. 2.2 („Arbeitsthesen“, S. 21)).

Die in diesem Zusammenhang relevanten kontextbezogenen Informationen, d. h. die Präferenzen des Nutzers, die Restaurants sowie die Entscheidungsalgorithmen, welche Restaurants als „passend“ anzusehen sind, werden im in Abb. 101 auf der nächsten Seite dargestellten Realweltuniversum des RestaurantFinders dargestellt. Die Funktionsweise dieses Kontextmodells erschließt sich bei dem im folgenden dargestellten schrittweisen Aufbau der Modellinstanz für den Anwendungsfall eines Touristen in Paris auf der Suche nach einem ortstypischen Restaurant mittlerer Preisklasse.

Im Ausgangszustand enthält die Modellinstanz lediglich ein die Anwendung repräsentierendes Topic sowie dessen Benutzer, wobei davon ausgegangen wird, dass die aus seinem Nutzerprofil stammenden Speisepreferenzen sowie sein Ort in Form geographischer Koordinaten bereits wie in Kap. 6.2.2 („Messenger-VUI“, S. 179) dargestellt integriert wurden (vgl. Abb. 102(a) auf der nächsten Seite, Abb. 96(a) auf S. 177).

Nun werden sämtliche Aktiven Assoziationen aktiviert, die neue Ausgangswerte erhalten haben. Wie in Abb. 102(b) auf der nächsten Seite dargestellt, ist dies zunächst nur die Aktive Assoziation *Nutzer in Stadt*, welche aus der geographischen Ortsangabe, die etwa im GPS-typischen WGS84-Format vorliegen

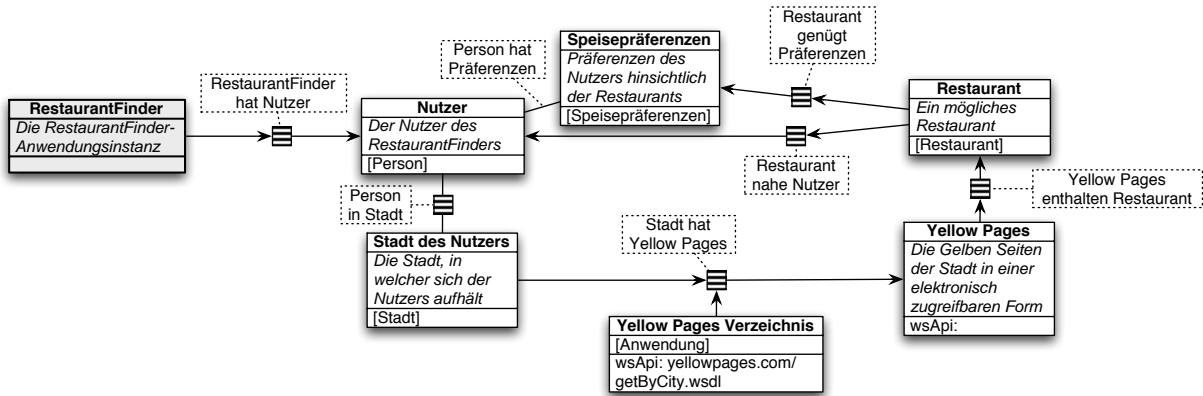
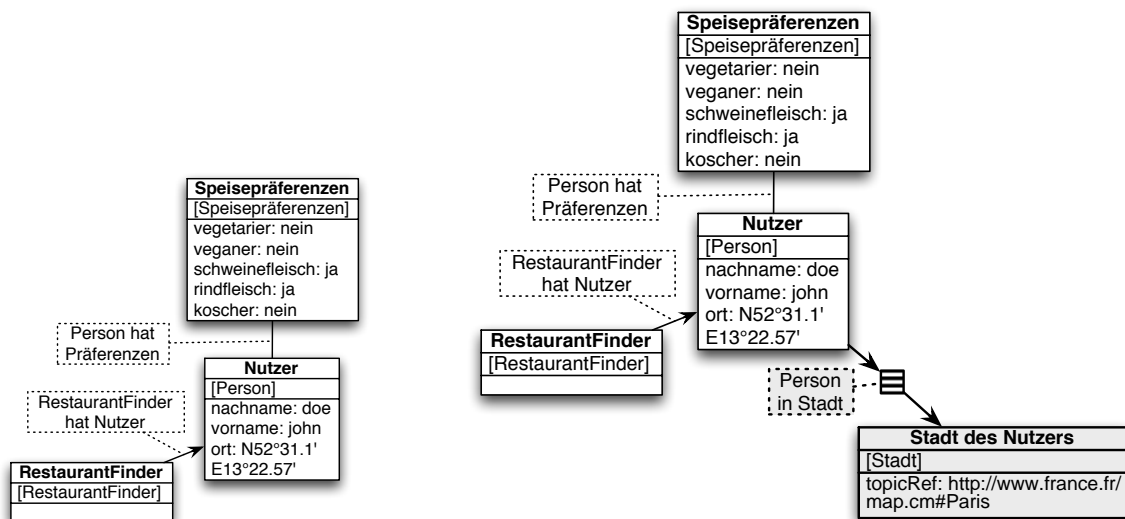


Abbildung 101: Universum der Restaurant-Finder-Anwendung



(a) Initialer Zustand der Modellinstanz

(b) Bestimmung des Nutzers sowie des Ortes

Abbildung 102: Schrittweiser Aufbau einer Modellinstanz der Restaurant-Finder-Anwendung

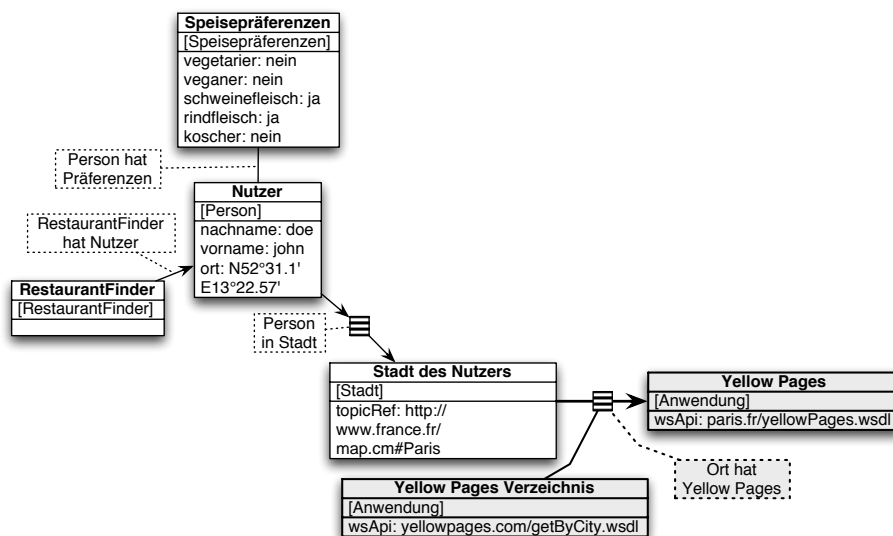


Abbildung 103: Bestimmung der Yellow Pages von Paris

kann, die betreffende Stadt bestimmt und deren PSI in das Topic *Stadt des Nutzers* einträgt. Dadurch können weitere Informationen über diese Stadt besorgt werden.

Das Update dieses Topics löst die Aktive Assoziation *Ort hat Yellow Pages* aus, welche wie in Abb. 103 dargestellt eine Abfrage bei dem durch das Topic *Yellow Pages Verzeichnis* repräsentierten Dienst vollzieht und auf diese Weise die Zugriffsschnittstelle für die Yellow Pages von Paris erhält, welche sie als Attribut in das Topic *Yellow Pages* einträgt. Die Aktualisierung dieses Topics löst wiederum die in Abb. 104 auf der nächsten Seite dargestellte Aktive Assoziation *Yellow Pages enthalten Restaurant* aus, welche eine Reihe von Topics des Typs *Restaurant* erzeugt, die alle Restaurants in Paris darstellen und einen Link auf deren Businesssysteme enthalten, die Informationen über deren Menü sowie die Anzahl freier Tische bereitstellen. Diese Informationen über die von den Wirtschaftssystemen bereitgestellten Web Services werden in den betreffenden Gelben Seiten vorausgesetzt.

An dieser Stelle ist der Informationsbeschaffungsprozess im Prinzip abgeschlossen, die in Abb. 104 auf der nächsten Seite dargestellte Kontextmodellinstanz enthält sämtliche benötigten Kontextquellen sowie die aus diesen gewonnenen Informationen. Die Mittel von *Context Maps* können jedoch noch weitergehend zur anwendungsspezifischen Modifikation bzw. Filterung der Informationsbasis verwendet werden, welche dann natürlich nicht mehr in der anwendungsübergreifenden Instanz, sondern einer anwendungsspezifischen Kopie des Weltmodells erfolgt und durch zusätzliche Aktive Assoziationen realisiert wird, welche die Map in der im folgenden dargestellten Weise umstrukturieren.

So führt das Erzeugen der Restaurant-Topics in Abb. 104 auf der nächsten Seite zur Ausführung der in Abb. 105 und 106 dargestellten Aktiven Assoziationen *Restaurant nahe Ort* und *Restaurant für Nutzer geeignet*, die überprüfen, ob sich das betreffende Restaurant in der Nähe des Nutzers befindet bzw. dessen Präferenzen erfüllt. Falls dies nicht der Fall ist, wird das betreffende Restaurant durch die entsprechende Aktive Assoziation aus der Map entfernt. Im konkreten Fall ist das Restaurant *Cafe Concorde* zu weit entfernt und *Elysee Kebap* entspricht nicht den Anforderungen des Nutzers, der ja nach einem ortstypischen Restaurant sucht. Auf diese Weise enthält die Map letzten Endes eine Reihe von für die

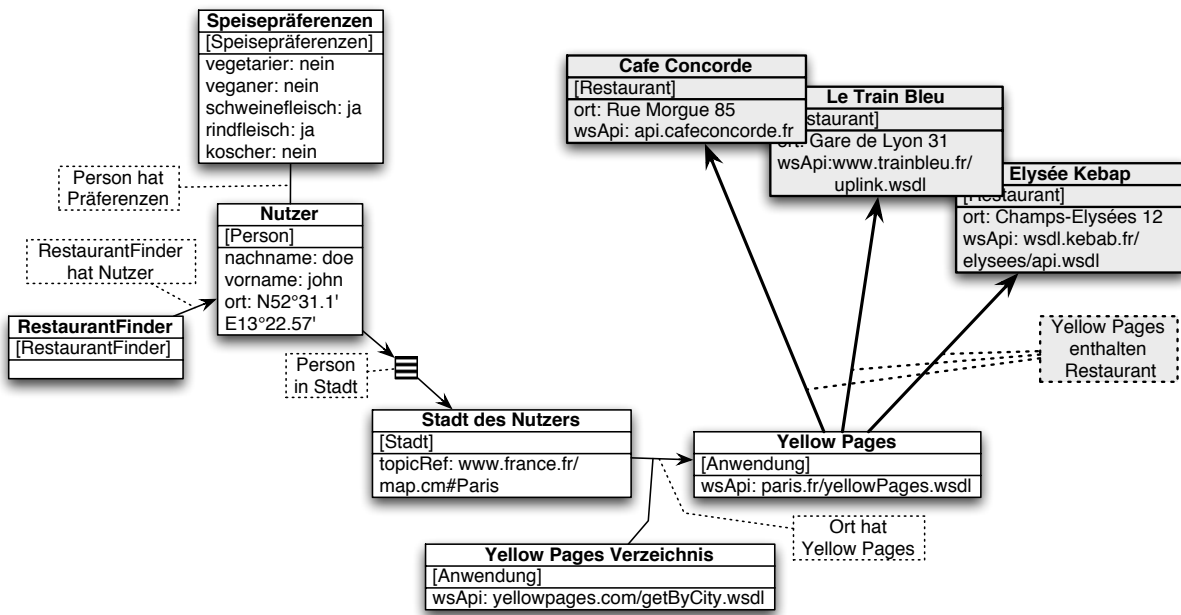


Abbildung 104: Bestimmung aller Restaurants in Paris

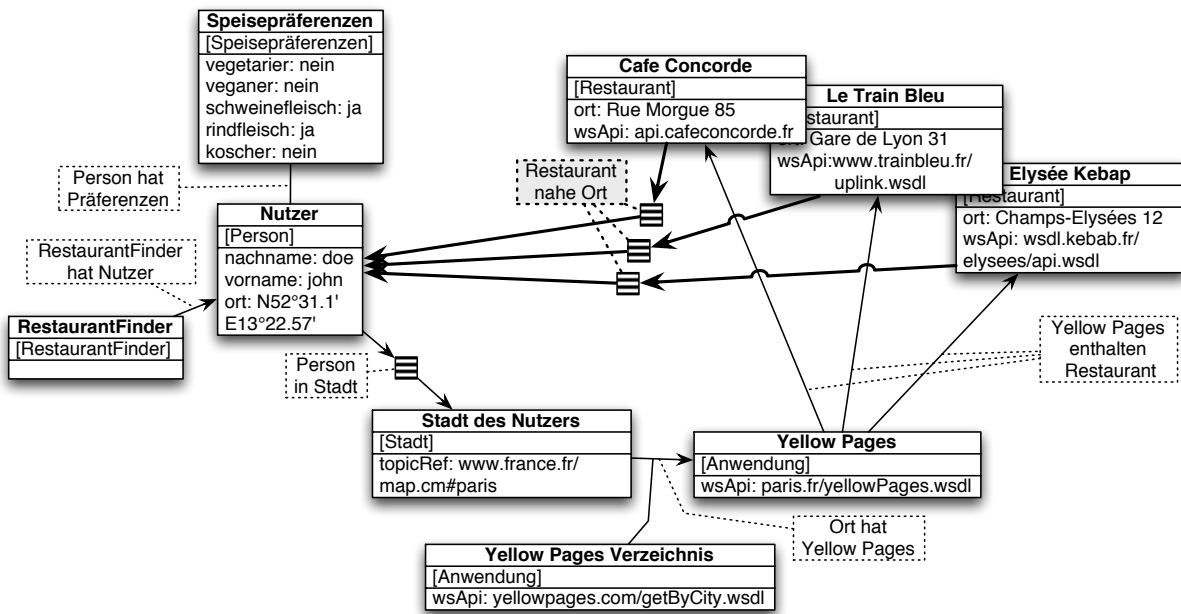


Abbildung 105: Herausfiltern der zu weit entfernten Restaurants

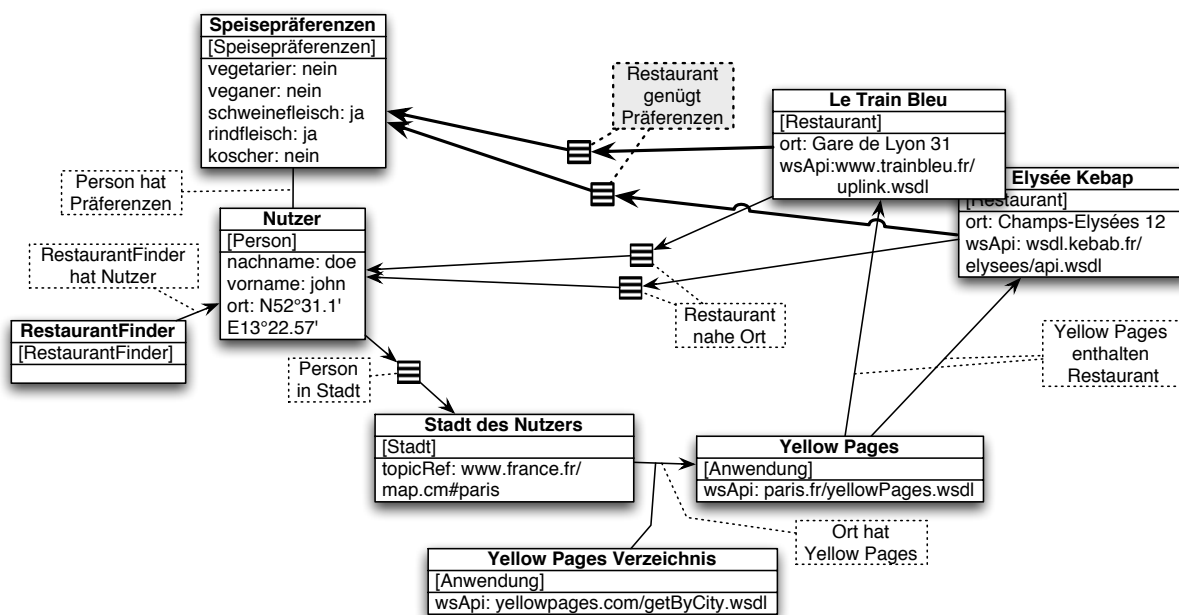


Abbildung 106: Entfernung der für die Nutzeranforderungen unpassenden Restaurants

gegebene Situation passenden Restaurants, welche von der Restaurant-Finder-Anwendung nur noch dargestellt werden müssen. Dieser endgültige Zustand des anwendungsbezogenen Kontextmodells ist in Abb. 107 auf der nächsten Seite dargestellt.

Hier zeigt sich ein großer Vorteil Aktiver Assoziationen. Es ist nicht notwendig, eine Assoziation zu jedem vom Nutzer präferierten Restaurant in Paris anzugeben, wie dies beim Topic-Map-Standard notwendig wäre, sondern es reicht, die Auswahlkriterien in Form eines Auswahlalgorithmus in einer Aktiven Assoziation zu formulieren, wodurch diese dann automatisch auf sämtliche relevanten Restaurants zeigt bzw. den Datenbestand sogar von unpassenden Informationen bereinigen kann.

Nach der optionalen Reservierung eines Tisches könnte die Anwendung eine entsprechende Information im persönlichen Datenbestand, etwa dem Terminkalender des Nutzers erzeugen, worauf die *Tourist Guide*-Anwendung oder das im folgenden vorgestellten kontextsensitiven Navigationssystem den Nutzer bei Bedarf zum Restaurant navigieren können.

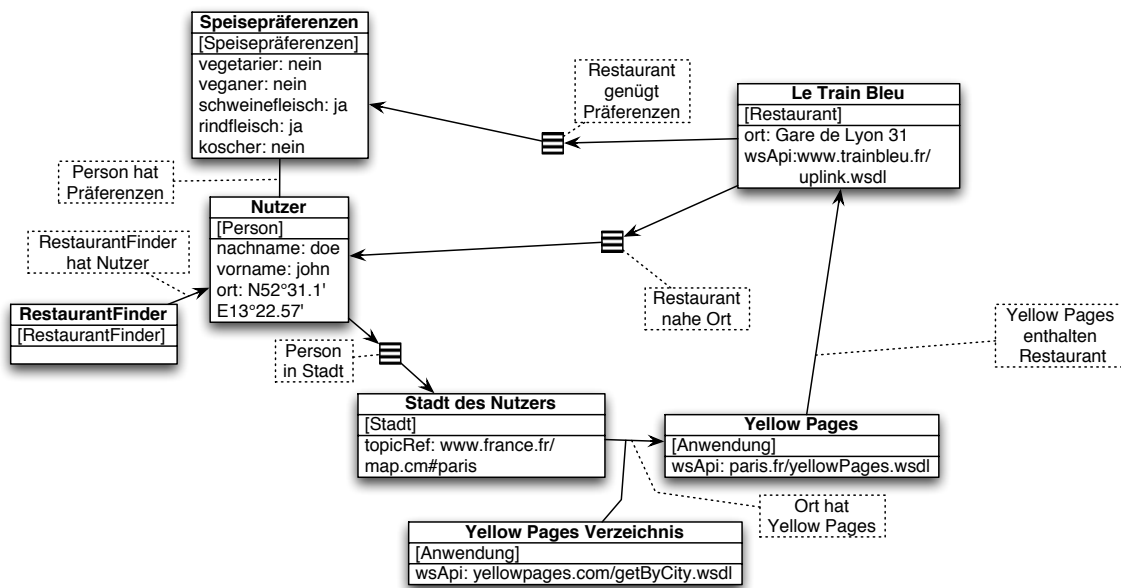


Abbildung 107: Endgültiger Zustand der Kontextmodellinstanz des Restaurant-Finders

6.2.4 Kontextsensitives Navigationssystem

Diese Anwendung stellt ein Beispiel mit einem komplexeren Kontextmodell dar, welches entsprechend vor allem die Konzepte zur Handhabung der Komplexität von Kontextmodellen mittels Abstrakter Assoziationen demonstriert.

Elektronische Navigationssysteme in KFZ geben dem Nutzer während der Fahrt Hinweise, wie das gewünschte Fahrtziel zu erreichen ist. Dazu müssen sie vor Fahrtantritt vom Nutzer über das gewünschte Fahrtziel informiert werden. Dies geschieht heute i. d. R. durch eine manuelle Eingabe von Ort, Straßennamen und ggf. Hausnummer mittels einer kleinen Tastatur am Gerät. Angesichts der Tatsache, dass der Nutzer persönlich keinen wirklichen Mehrwert aus der Kenntnis genauer Straßennamen und Postleitzahlen seiner Fahrtziele zieht und daher nur an deren *Nutzung*, nicht jedoch deren *Kenntnis* interessiert ist, sowie dass Informationen über mögliche Fahrtziele heute immer häufiger mittels elektronischer Informationssysteme wie Adressbücher oder Notizfunktionen in mobilen Computersystemen verwaltet werden, stellt der Vorgang der Recherche der Zieladresse in einem mobilen elektronischen Organizer und der anschließenden manuellen Eingabe der gefundenen Informationen in das Navigationssystem einen unnötigen Medienbruch dar, den es durch geeignete technische Systeme zu überwinden gilt. Daher wurde die *Software for Context-Aware Navigation* (SCANav) entwickelt, welche aus den persönlichen Informationen des kontextsensitiv bestimmten Nutzers mögliche Fahrtziele ableitet. Diese Fahrtziele werden dem Benutzer vor Fahrtantritt in einem Display zur Auswahl angeboten, worauf der Nutzer eines auswählt und sofort die Fahrt beginnen kann, ohne in dieser Situation mit seinen Adressbüchern, PDAs, ausgedruckten E-Mails oder gar seinem Notebook hantieren zu müssen.

SCANav bietet zwei Benutzungsmodi. Im *manuellen Modus* kann der Nutzer seine Termine, E-Mails und Kontakte einsehen und darin das Fahrtziel manuell auszuwählen. Dies ist in Abb. 108 auf der gegenüberliegenden Seite beispielhaft anhand einer E-Mail, welche im Text die gewünschte Adresse enthält,

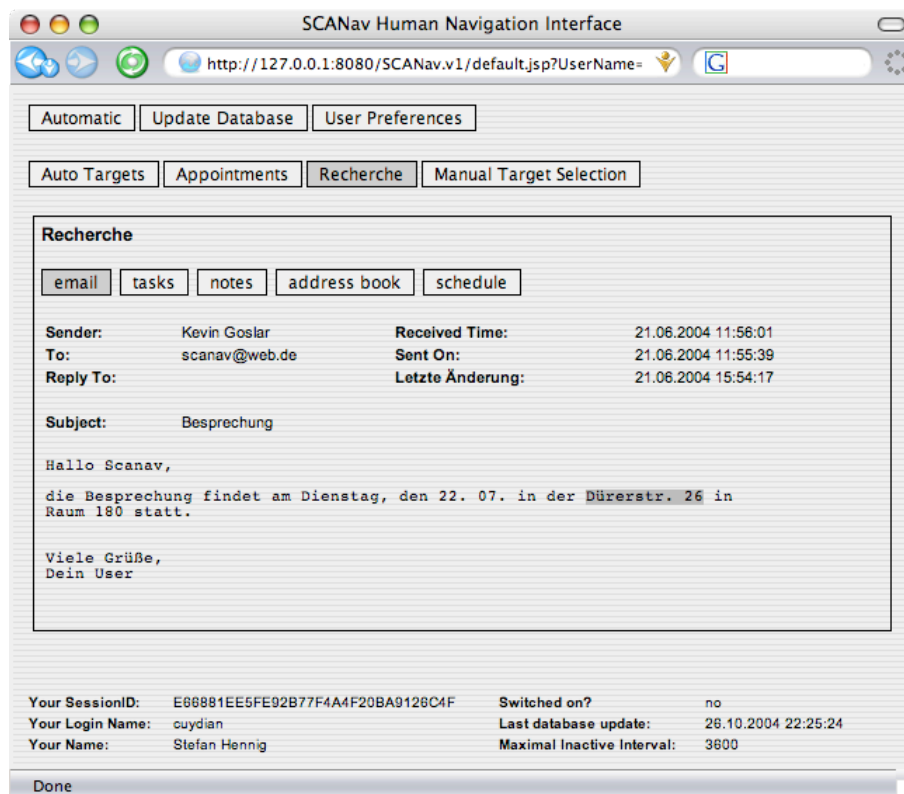


Abbildung 108: Manuelle Selektion des Fahrtziels im Text einer EMail

dargestellt. Dazu zählt auch die Funktion, eine Volltextsuche im gesamten bekannten Datenbestand durchzuführen. Abb. 109 auf der nächsten Seite zeigt dazu die Ergebnisse einer Suche nach dem Wort „Kevin“. Ebenfalls möglich ist die Eingabe des Namens einer Person, deren diverse Adressen daraufhin im Adressbuch des Nutzers nachgeschlagen und als Fahrtziel verwendet werden. Gleiches gilt für ihren aktuellen Aufenthaltsort, der aus ihrem Nutzerprofil bestimmt wird.

Im *automatischen Modus* bestimmt die Anwendung hingegen selbsttätig eine Liste von Fahrtzielkandidaten aus den nächsten anstehenden Terminen und präsentiert sie dem Nutzer zur Auswahl (vgl. Abb. 110 auf der nächsten Seite). Um den Vorgang der Bestimmung der Fahrtziele des Nutzers möglichst weitgehend zu automatisieren, ist es notwendig, eine Reihe von externen Informationsquellen bezüglich der Identität des aktuellen Nutzers sowie dessen Plänen zu befragen. Tab. 25 auf Seite 190 stellt ein paar diesbezügliche Kontextquellen dar.

Das Kontextmodell wird selbstverständlich durch *Context Maps* realisiert, die wie bereits bei der *Restaurant-Finder-Anwendung* dargestellt zur Laufzeit schrittweise aus den verfügbaren Informationen aufgebaut und vervollständigt werden. Abb. 111 auf Seite 189 zeigt das gesamte Kontextmodell. Es ist trotz der relativ geradlinigen Funktion von SCANav bereits hinreichend umfangreich, außerdem bedingen die vielfältigen Zusammenhänge zwischen den einzelnen Rollen eine intensive und unübersichtliche Vernetzung ihrer Repräsentationen im Modell. Durch den gezielten Einsatz von Abstrakten Assoziationen kann dieses Kontextmodell jedoch Schritt für Schritt modelliert und interpretiert werden.

Abb. 112 auf Seite 190 zeigt zunächst eine sehr stark abstrahierte *high-level*-Darstellung des Kontextmodells, aus welcher ersichtlich wird, dass die Anwendung die möglichen Fahrtziele bestimmt, indem

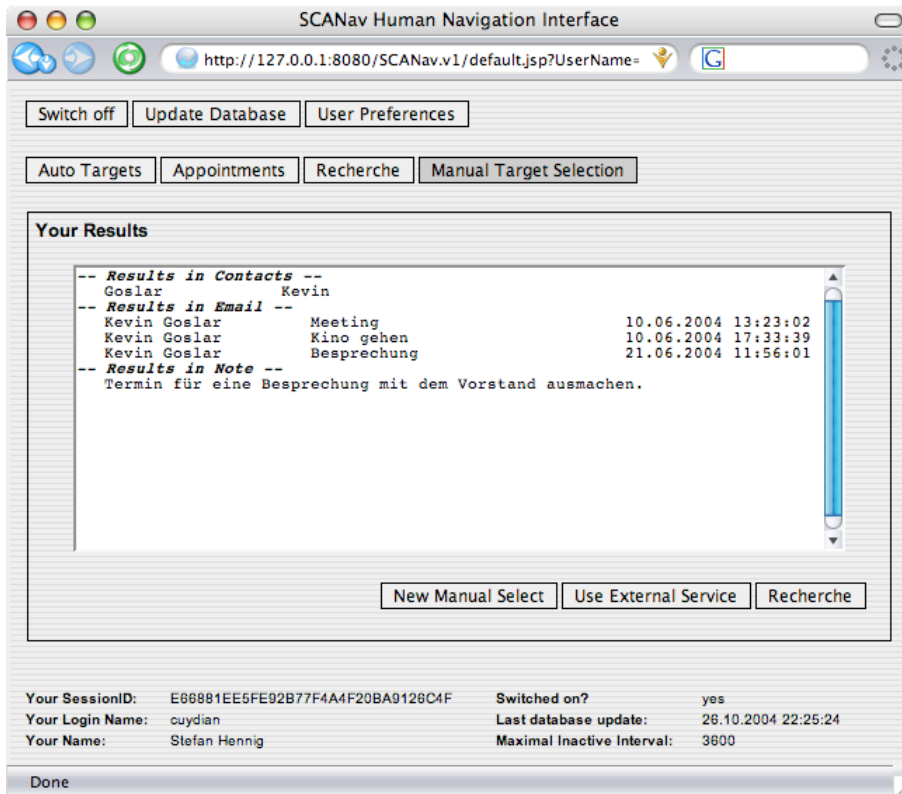


Abbildung 109: Volltextsuche nach "kevin" im Datenbestand des Nutzers

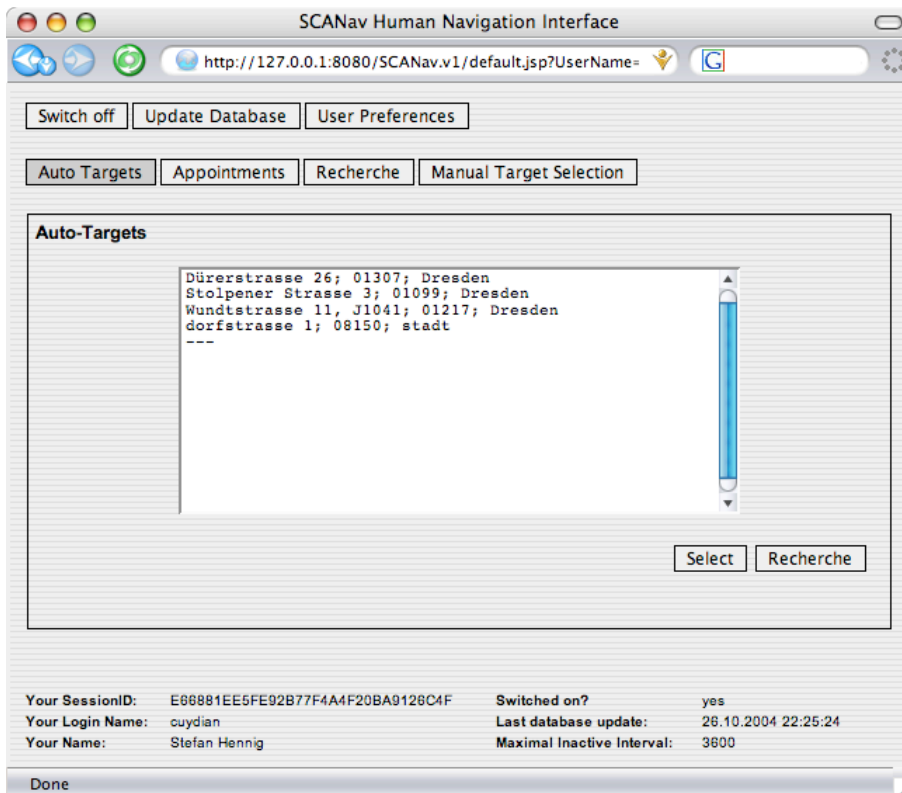


Abbildung 110: Liste der automatisch bestimmten Fahrtziele des Nutzers

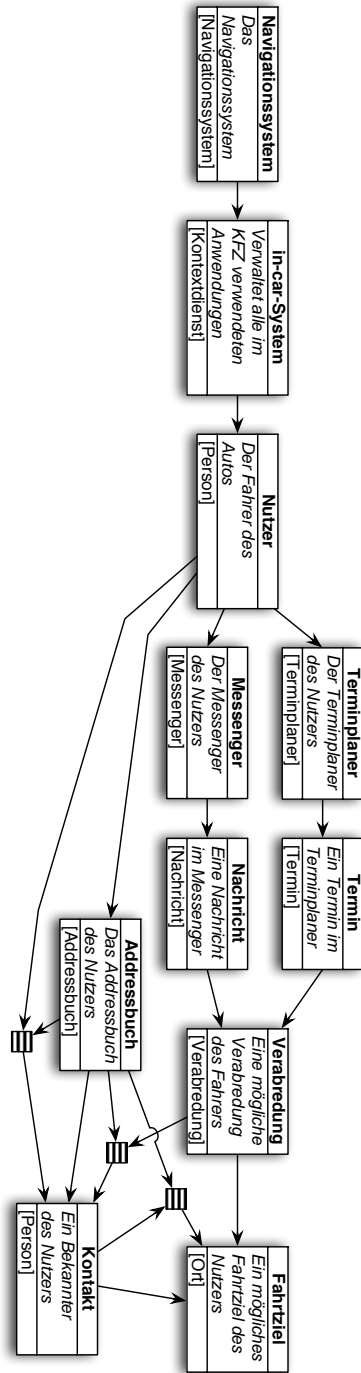


Abbildung 111: Gesamte Darstellung des SCANav-Realweltuniversums

Kontextquelle Beschreibung	
in-car Kontextdienst	Ein innerhalb des Fahrzeugs befindlicher Kontextdienst ermittelt die Identität des Fahrers. Dafür wird die im Auto vorausgesetzte biometrische Erkennung des Fahrers etwa im Rahmen einer Diebstahlsicherung verwendet.
Terminplaner	gibt Aufschluß über die vom Nutzer geplanten Verabredungen, woraus sich diverse Fahrtziele ableiten lassen,
Messaging-Anwendung	kann Verabredungen in textueller Form, etwa im EMail-Verkehr des Nutzers enthalten,
Adressbuch	gibt Aufschluß über die Bekannten des Nutzers sowie deren diverse Adressen.

Tabelle 25: Von einem kontextsensitiven Navigationssystem verwendete Kontextquellen

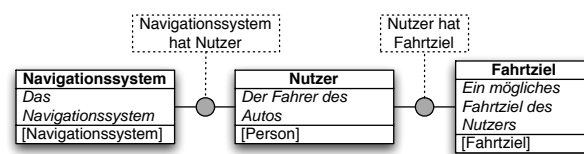


Abbildung 112: High-level-Sichtweise des Kontextmodells des Navigationssystems

sie zunächst die Identität ihres Nutzers, d. h. des Fahrers und anschließend dessen Fahrtziele bestimmt. Diese Abbildung kann als eine aufgeklappte Abstrakte Assoziation *Navigationssystem hat Fahrtziel* aufgefasst werden, welche die abstrakteste Sichtweise auf dieses Kontextmodell darstellen könnte, an dieser Stelle jedoch nicht dargestellt wird.

Abb. 113 zeigt den ersten Teil des Kontextmodells in der aufgeklappten Darstellung. Mittels der Aktiven Assoziation *Navigationssystem sucht in-car-System* wird der in-car-Kontextdienst bestimmt und dieser mittels der Aktiven Assoziation *in-car-System bestimmt Fahrer* über die Identität des aktuellen Fahrers befragt, welche wie bereits erwähnt eine URI auf dessen Nutzerprofil enthält, die in das Topic *Fahrer* eingetragen wird.

Das weitere Vorgehen wird in Abb. 114 auf der gegenüberliegenden Seite dargestellt, in welcher die Abstrakte Assoziation *Fahrer hat Fahrtziel* aus Abb. 112 und damit der Teil zur Bestimmung den Fahrtziele des Fahrers aufgeklappt und der eben besprochene Teil wieder zugeklappt ist. Um die Fahrtziele des Fahrers zu bestimmen, gibt es wie dargestellt zwei Wege. Der eine besteht darin, die Verabredungen des Fahrers zu bestimmen, um daraus potentielle Fahrtziele zu gewinnen. Der andere besteht darin, dass der Fahrer selbst einen Kontakt angibt, zu dem er fahren möchte. Die Details zum ersten Weg werden in Abb. 115 auf der gegenüberliegenden Seite dargestellt. Um die Darstellung möglichst ein-

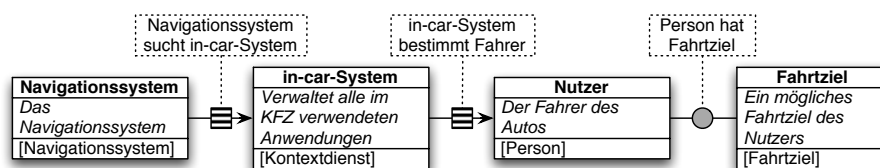


Abbildung 113: Low-Level-Sichtweise auf die Bestimmung des Fahrers

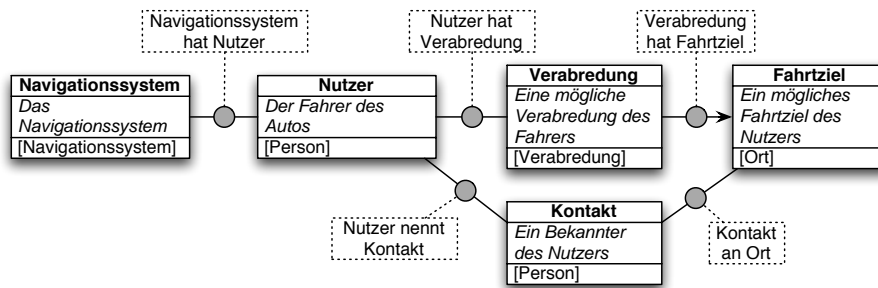


Abbildung 114: Bestimmung der Fahrtziele des Fahrers

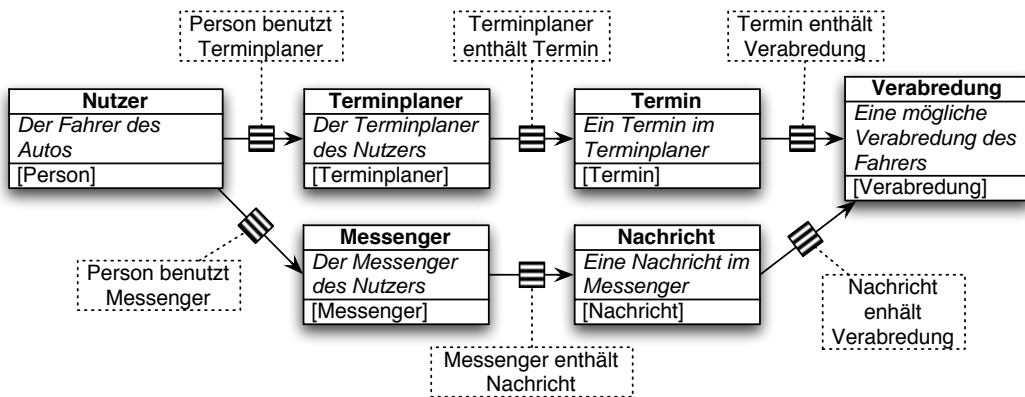


Abbildung 115: Low-level-Map der Assoziation *Nutzer hat Verabredung*

fach zu halten, wird an dieser Stelle nur die *low-level*-Sichtweise der Abstrakten Assoziation *Nutzer hat Verabredung* aus Abb. 114 dargestellt. Dazu werden zunächst der *Terminplaner* sowie der bereits bekannte *Messenger* bestimmt, um in den darin enthaltenen Terminen und E-Mails nach Tätigkeits-, Orts- und Zeitangaben zu suchen, die als *Verabredung* interpretiert werden. Dies geschieht durch eine Art von *Text Mining*, worunter das Extrahieren von Informationen aus natürlichsprachlichen Texten verstanden wird (vgl. [FeDa95, Fay⁺96]). Da ein Verständnis des kompletten Textinhaltes beim heutigen Stand der Technik auch für die absehbare Zukunft nicht vorausgesetzt werden kann, wird sich dabei auf die Extraktion sehr einfacher Informationsarten beschränkt, die dafür mit einer hohen Zuverlässigkeit erfolgt. Im vorliegenden Fall werden vereinfachend reguläre Ausdrücke zur Identifikation der benötigten Informationen verwendet, was sich als praktikabel erwiesen hat, wie in Abb. 116 auf der nächsten Seite zu sehen ist.

Abb. 117 auf der nächsten Seite zeigt das Kontextmodell mit aufgeklappter Assoziation *Verabredung hat Fahrtziel*, welche offenbart, dass als Fahrtziel einer Verabredung nicht nur der Ort der Verabredung verwendet wird, was durch die Aktive Assoziation *Verabredung enthält Fahrtziel* realisiert wird, sondern auch die Orte der bei der Verabredung beteiligten Kontaktpersonen, d. h. deren Wohnort, Arbeitsstelle oder augenblicklicher Aufenthaltsort.

Zum Schluß bleibt noch zu klären, wie die als Fahrtziel geeigneten Orte einer Person bestimmt werden. Dieser Zusammenhang tritt in Form der Abstrakten Assoziation *Kontakt an Ort* in Abb. 114 und 117 auf. Abb. 118 auf Seite 193 zeigt deren *low-level-Map*, die darstellt, dass sowohl der augenblickliche Aufenthaltsort, bestimmt mittels *Kontakt an Ort*, als auch die im Adressbuch des Nutzers vermerkten

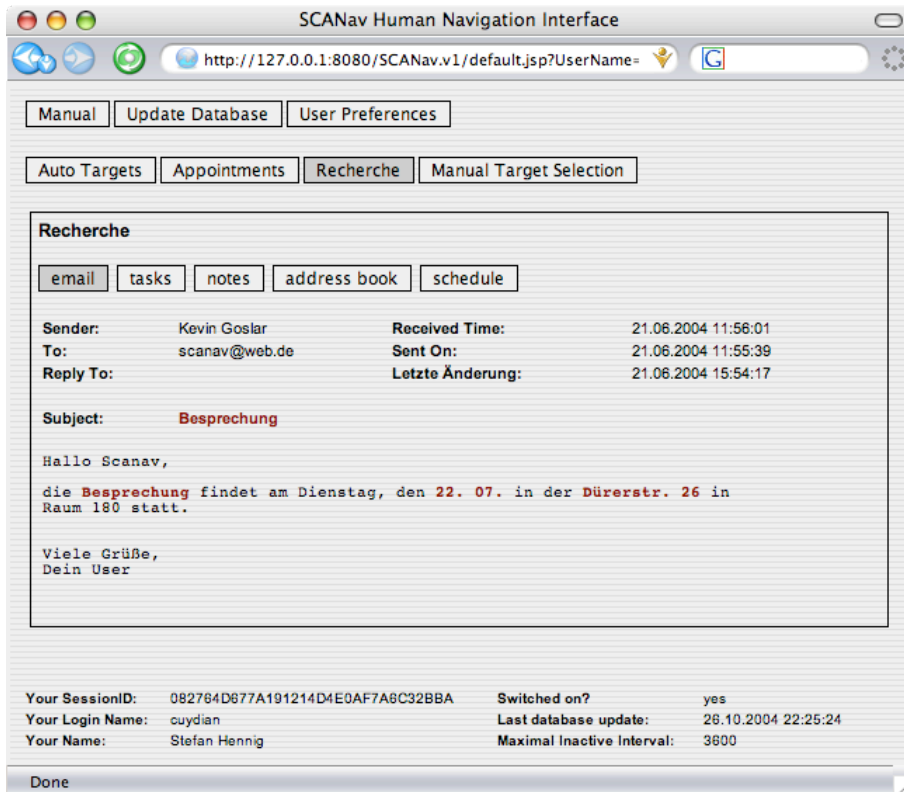


Abbildung 116: Von SCANav per Text-Mining bestimmte Informationen in einer EMail

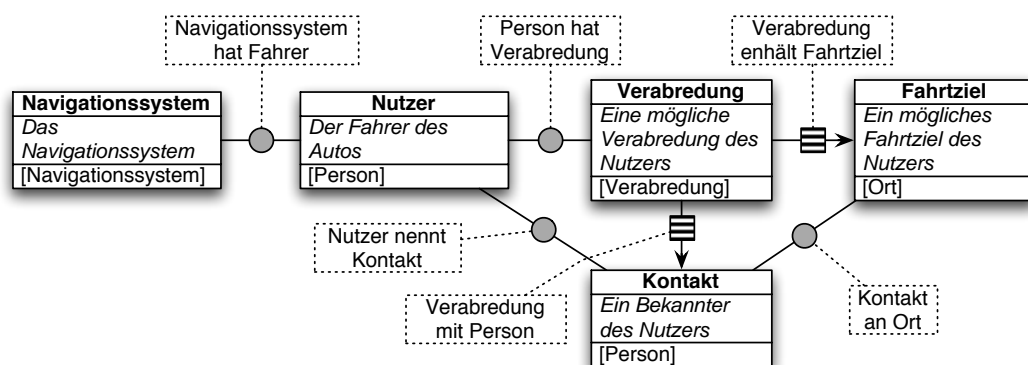
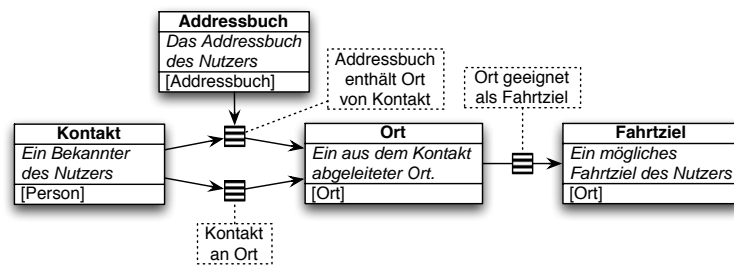


Abbildung 117: Kontextmodell mit aufgeklappter Assoziation Verabredung hat Fahrtziel

Abbildung 118: Low-level-Map der Abstrakten Assoziation *Kontakt an Ort*

Orte dieser Person, bestimmt durch *Adressbuch enthält Ort von Kontakt* dafür berücksichtigt werden, allerdings nur dann als *Fahrtziel* Verwendung finden, wenn sie durch die Aktive Assoziation *Ort geeignet als Fahrtziel* auch als sinnvolles *Fahrtziel* bewertet wurden. Die gleiche Bewertung wird im übrigen auch von der Aktiven Assoziation *Verabredung enthält Fahrtziel* in Abb. 117 vorgenommen.

Damit ist das Kontextmodell der SCANav-Anwendung vollständig beschrieben, was dank der dazu verwendeten Abstrakten Assoziationen in einer einfachen, übersichtlichen und trotzdem detaillierten und vor allem zusammenhängenden Form erfolgen konnte.

Zum Zugriff auf die Informationen in externen Anwendungen werden *Zugriffsmodule* verwendet. Implementiert wurde ein Zugriffsmodule auf Microsoft Outlook, da dort sämtliche von dieser Anwendung benötigten Informationstypen enthalten sind. Dieses Zugriffsmodule greift Outlook über dessen API für Visual Basic zu und stellt seine Dienste mittels einer *Web Service*-Schnittstelle in einer standardisierter Form zur Verfügung. Analog funktionierende und in gleicher Weise verwendbare Zugriffsmodule können für beliebige andere Kontextquellen realisiert werden.

Schlußendlich könnte sich das Navigationssystem selbst als für den Nutzer verfügbare Anwendung in dessen Nutzerprofil eintragen, wodurch seine Dienste auch durch andere Anwendungen verwendet werden könnten, etwa den im folgenden vorgestellten *Tourist Guide*.

6.2.5 Tourist Guide

Eine der ersten kontextsensitiv erfassten Kontextdimensionen stellte der physikalische *Ort* des Nutzers dar, entsprechend waren die ersten kontextsensitiven Anwendungen vor allem ortssensitiv, was oft auch als *Location-Based Services* umschrieben wird. Eine der prominentesten Anwendungen in diesem Bereich stellen die inzwischen recht zahlreich beforschten Touristenführungssysteme dar, die – auf einem ortssensitiven PDA laufend – einen ortsunkundigen Touristen durch eine fremde Stadt geleiten und etwa Informationen über die Gebäude, vor denen der Nutzer steht, bereitstellen. Beispiele sind im weiteren Sinne das *Cooltown*-Projekt der HEWLETT-PACKARD LABORATORIES (vgl. [BaKi01]), das *Cyberguide*-System von ABOWD (vgl. [Abo⁺97]), das *GUIDE*-System (vgl. [Che⁺00]) sowie der *Tour Guide* der LANCASTER UNIVERSITÄT (vgl. [Dav⁺01]).

Diesen bereits existierenden Anwendungen soll als abschließendes Anwendungsbeispiel ein *Tourist Guide* genannter Touristenführer gegenüber gestellt werden, welcher die Informationen und Dienste der anderen in diesem Kapitel vorgestellten Anwendungen miteinander kombiniert und dadurch weitergehende Dienste für Touristen anbietet. Dazu gehört die Navigation des Nutzers mittels des jeweils opti-

Quelle	Beschreibung
SCANav	Navigation des PKW zu den vom <i>Tourist Guide</i> elektronisch übermittelten Koordinaten
ÖPNV	Auskunft über die geeignetste Verbindung zwischen zwei Orten in der Stadt, Fahrpläne und Preise, ggf. auch Buchungssystem für Tickets
Autovermietung	gibt Auskunft über das Auto des Nutzers und dessen Komponenten, etwa unter welcher IP-Adresse das Navigationssystem dieses Autos zu finden ist
Nutzerprofil	enthält die benötigten Bezahlinformationen, entweder in Form der Konto- oder Kreditkartennummer des Nutzers, eines Micropaymentsystems oder in Form eines Links auf dessen Hotel, über welches die anfallenden Kosten abgerechnet werden.

Tabelle 26: Vom *Tourist Guide* verwendete Kontextquellen

malsten Verkehrsmittels, wobei die Anfahrt des Nutzers in die Stadt mittels seines PKW erfolgen kann, während er sich innerhalb der Stadt aufgrund der dort oft problematischen Verkehrsverhältnisse dann vorwiegend mittels öffentlicher Verkehrsmittel bzw. zu Fuß bewegen wird. Dabei soll er in sämtlichen Fällen von einem vom *Tourist Guide* gesteuerten Navigationssystem unterstützt werden, d. h. ihm wird mitgeteilt, wann er wo abbiegen bzw. in welches Verkehrsmittel er ein- und aussteigen soll. Da der *Tourist Guide* die Route des Nutzers kennt, kann er eine optimale Entscheidung über das zu verwendende Verkehrsmittel treffen. Ebenfalls soll der *Tourist Guide* wenn möglich die zur Benutzung des jeweiligen öffentlichen Verkehrsmittels notwendigen Tickets buchen. Dadurch ist der Nutzer in der Lage, sich selbst bei problematischen Verkehrsverhältnissen dank des Touristenführers frei und sicher durch eine fremde Stadt zu bewegen, wobei er aber nicht auf Flexibilität und Spontaneität verzichten muß, da er ja nach wie vor gehen kann, wohin er will.

Der *Tourist Guide* greift auf die in Tab. 26 dargestellten Kontextquellen zu. SCANav stellt dabei das bereits besprochene, im Nutzerprofil als verfügbare Anwendung verzeichnete Navigationssystem im KFZ des Nutzers dar. Die heute i. d. R. noch hypothetische Web Service-Schnittstelle der Verkehrsbetriebe der jeweiligen Stadt dient dazu, das jeweils geeignetste Verkehrsmittel und dessen Fahrpläne abzurufen sowie ggf. gleich elektronisch Tickets für den Nutzer zu buchen. Ebenfalls verwendet wird das in Kap. 6.2.1 („Nutzerprofil“, S. 176) beschriebene und in Abb. 96(a) auf S. 177 dargestellte Lokalisierungssystem des Nutzers, das demzufolge ebenfalls nicht vom *Tourist Guide* implementiert werden muß.

Abb. 119 auf der gegenüberliegenden Seite zeigt das Realweltuniversum des Kontextmodells des *Tourist Guide*. Jede der dargestellten Rollen beinhaltet eine Fülle von Informationen aus einer jeweils anderen externen Informationsquelle. Diese heterogenen Informationen werden durch die Context Map in einer sehr effektiven und transparenten Weise zu einem umfassenden, anwendungsübergreifenden, aber trotzdem stets akkuraten und aktuellen Kontextmodell vereinigt. Die beschriebene Funktionalität könnte zweifelsohne bereits heute realisiert werden, allerdings würde eine solche Lösung eine teure Spezialanwendung darstellen, die in den meisten Fällen ökonomisch nicht zu rechtfertigen wäre, da ein wesentlicher Teil der Erstellungs- und Wartungskosten für die Module zur Beschaffung der Informationen aus den heterogenen Kontextquellen aufzuwenden wäre. Es ist einsichtig, dass eine solche Anwendung ba-

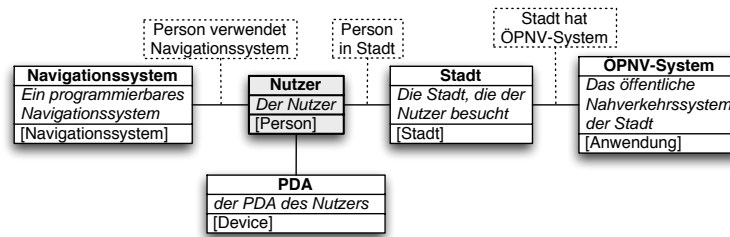


Abbildung 119: Template des *Tourist Guide*

sierend auf einem in Abb. 119 dargestellten Kontextmodell viel einfacher zu erstellen und auch einfacher zu warten wäre, da beim Hinzukommen neuer Kontextquellen, etwa einer neuen Autovermietung, lediglich einige Anpassung in der betreffenden Context Map, nicht jedoch im Anwendungscode des Tourist Guide notwendig sind. Der Tourist Guide selbst stellt dadurch letztlich nur noch eine relativ kompakte Anwendung dar, die gezielt bereits existierende Anwendungen bzw. deren Informationen zugreift bzw. steuert und so eine sehr umfassende und komplexe Funktionalität realisiert. Die konkrete Anwendungslogik des *Tourist Guide* ist an dieser Stelle von keinem thematischen Interesse und wird daher nicht besprochen.

Der Punkt, auf den dieses Kapitel abzielt, ist die Tatsache, dass durch die Nutzung des in dieser Arbeit entwickelten systematischen und komfortablen Integrations- und Darstellungsmodells für kontextbezogene Informationen aus heterogenen Kontextquellen diese Kontextinformationsbestände zum einen überhaupt erst einmal in der beschriebenen umfassenden Weise nutzbar werden und dies zum anderen auch noch in einer sehr einfachen und transparenten Art und Weise möglich wird. Damit stellen Context Maps einen essentiellen Schritt auf dem Weg zur tatsächlichen Realisierung fortgeschrittener kontextsensitiver Konzepte, d. h. auf deren Weg vom Labor zum Marktplatz dar. Dieser Weg ist bisher trotz jahrelang vorliegender Forschungsergebnisse noch nicht oder nur sehr ansatzweise gegangen worden, was zu einem erheblichen Teil an den beschriebenen ökonomischen Ursachen liegt.

6.3 Konzeptionelle Validierung

Neben der bereits durchgeführten technischen und funktionalen Validierung wird an dieser Stelle die Erreichung der wissenschaftlichen Zielstellungen der Arbeit bewertet.

6.3.1 Wissenschaftliches Ergebnis

Hinsichtlich des *Informationsmanagements* hatte diese Arbeit die Entwicklung eines generischen Darstellungs- und Integrationsmodells für Kontextinformationen, welches Kontextverbraucher bei der Integration heterogener und verteilter Kontextinformationen in einer strukturierten Weise unterstützt, zum Ziel. Dies geschieht im Rahmen von Bemühungen, eine robuste, d. h. dauerhaft verfügbare und qualitativ bestmögliche Kontextinformationsbasis als notwendige Grundlage für ein verlässliches Verhalten von kontextsensitiven Unternehmensprozessen zu realisieren. Konkret wurde dafür ein Modell entwickelt, welches eine Menge von externen Kontextinformationen in einen mittels Ontologien definierten und durch szenische Beschreibungen wiederverwendbar strukturierten anwendungs- und quellenüber-

greifenden Bedeutungszusammenhang einordnet und als umfassendes Realweltmodell zur Benutzung durch Kontextinformationsverbraucher darstellt.

Dabei wird das Detailwissen über die Identität, Schnittstellen und Datenmodelle der einzelnen Kontextquellen in Form eines strukturierten Informationszugriffs- und -integrationsmodells expliziert, welches wiederum als Basis für ein Darstellungsmodell für kontextbezogene Informationen fungiert und so durch maschinelle Aufgabenträger im sozio-technischen Kontextinformationssystem, welches einen Teil des betrieblichen Informationssystems darstellt, unterstützbar wird. Als Folge davon kann der Umfang des von den Verbrauchern der Kontextinformationen benötigten Wissens und Aufwandes zur Beschaffung der von ihnen benötigten Kontextinformationen signifikant reduziert werden, indem Anwendungssysteme eingesetzt werden, welche dieses Kontextinformationsintegrationsmodell realisieren und so die benötigten Kontextinformationen in einer singulären, homogenen, strukturierten sowie semantisch aussagefähigen Form innerhalb eines einzigen, sämtliche Anwendungsbereiche umfassenden Realweltmodells verfügbar machen.

In *wirtschaftsinformatischer* Hinsicht stellt diese Arbeit neben der Darstellung des auf der betriebswirtschaftlichen Ebene existierenden Bedarfes nach preiswerten und umfassenden Kontextinformationen als Motivation für den eben beschriebenen kontextinformationsmanagementbezogenen Ansatz, welcher den Schwerpunkt dieser Arbeit darstellt, eine Einordnung der in dieser Arbeit betrachteten kontextbezogenen Konzepte in ebendieses betriebswirtschaftliche Umfeld dar.

In *betriebswirtschaftlicher* Hinsicht konnte daher festgestellt werden, dass die durch Context Maps ermöglichte effektivere und effizientere Gestaltung des Kontextinformationsmanagements, d. h. der vereinfachte und verbilligte Kontextinformationsbeschaffungsprozess als betriebliche Querschnittsfunktion eine ganze Palette wünschenswerter betriebswirtschaftlicher Effekte bewirkt, wozu die ebenfalls effektivere und effizientere Gestaltung vieler betrieblicher Funktionen sowie darauf basierend die Möglichkeit, neuartige oder zumindest in ihrer Funktionalität signifikant erweiterte betriebliche Geschäftsanwendungen, Leistungsprozesse und damit letztlich auch betriebliche Wertschöpfungskonzepte zu realisieren zählen.

6.3.2 Anforderungsanalyse

Eine Beurteilung des zur Integration heterogener Kontextinformationen notwendigen Funktionsumfangs wurde mittels des in Kap. 2.4.1 („Bewertungsschema“, S. 25) entwickelten Bewertungsschemas sowohl bei den in Kap. 4 („Verwandte Arbeiten“, S. 97) beschriebenen verwandten Arbeiten als auch bei den in dieser Arbeit entwickelten *Context Maps* vorgenommen.

An dieser Stelle sollen nur die Ergebnisse dieser Analyse vorgestellt werden, eine detaillierte Diskussion der einzelnen Anforderungen findet sich wie gehabt in Anhang A.14 auf S. LXIV.

Wie aus den in Abb. 120 auf Seite 198 dargestellten Graphen ersichtlich ist, erfüllen *Context Maps* sämtliche Anforderungen in einer signifikant besseren Weise als die verwandten Arbeiten. Zum einen ist ihre in Abb. 120(a) dargestellte durchschnittliche Bewertung der einzelnen Anforderungskomplexe jeweils zwischen 4,3 und 5, zum anderen wird nie ein Ergebnis schlechter als mit 4 bewertet (vgl. Abb. 120(b)). Ebenfalls markant unterscheiden sich Context Maps bezüglich der Anzahl der mit voller Punktzahl (5 Punkten) bewerteten Anforderungen, welche in Abb. 120(c) dargestellt sind, von den verwandten

Arbeiten. Dabei ist anzumerken, dass die Bandbreite des Bewertungsspektrums des verwendeten Bewertungsschemas die zusätzlichen Funktionalitäten von *Context Maps* nicht immer angemessen widergespiegelt, d. h. *Context Maps* erfüllen die gegebenen Anforderungen oft in einer wesentlich geeigneteren Weise als andere, ebenfalls mit 5 Punkten bewertete Arbeiten. Auf die Verteilung von Zusatzpunkten für *Context Maps* wurde jedoch verzichtet, da dieser Schritt eine Spezialisierung des Bewertungsmodells auf die Funktionalität von *Context Maps* bedeutet hätte, was jedoch explizit vermieden werden sollte.

Als Ergebnis der Anforderungsanalyse von *Context Maps* kann festgestellt werden, dass diese im Gegensatz zu sämtlichen dargestellten verwandten Arbeiten die geforderten Funktionalitäten nicht nur vollständig, sondern oft auch signifikant besser erfüllen. Insgesamt decken sie sämtliche geforderten Anforderungsbereiche in einer gut geeigneten, sehr oft sogar sehr geeigneten Art und Weise ab. Damit können *Context Maps* als die erste umfassende Lösung für das in dieser Arbeit betrachtete Verteilungs- und Integrationsproblem von kontextbezogenen Informationen angesehen werden.

6.4 Problemdiskussion

Aus der Nutzung von umfassenden Kontextinformationen ergeben sich neben den bereits dargestellten zahlreichen positiven Effekten auch eine Reihe von Problemen, die im folgenden skizziert werden. Diese Probleme können sich auf die Akzeptanz und damit die Einsatzmöglichkeiten von *Context Maps* auswirken und werden daher an dieser Stelle im Sinne einer Validierung der praktischen Einsatzfähigkeit von *Context Maps* diskutiert. Indem zu sämtlichen an dieser Stelle geäußerten Problemen Auswege skizziert werden, wird die Einsatzfähigkeit des in dieser Arbeit entwickelten Prinzips im praktischen Einsatz begründet und untermauert.

6.4.1 Unmöglichkeit der Nutzenquantifizierung

Es dürfte sehr schwer bzw. sogar unmöglich sein, den durch *Context Maps* erbrachten Nutzen mittels einer konkreten Zahl, etwa in monetärer Hinsicht, zu beziffern, denn die Verwendungsmöglichkeiten und damit der Nutzen von Kontextinformationen in betrieblichen Anwendungssystemen sind vielfältig, da Kontextsensitivität eine betriebliche *Querschnittsfunktion* darstellt. Zusätzlich stellt die in dieser Arbeit entwickelte Lösung einen *gesamtheitlichen* Ansatz dar, der erst bei der Kombination vieler Kontextquellen in komplexen Szenarien sinnvoll einsetzbar wird, was die Berechnung des dadurch an vielen Stellen auftretenden Gesamtnutzens weiter verkompliziert. Weiterhin kann eine auf *Context Maps* basierende Lösung immer nur so gut wie die Informationen in den verfügbaren *Kontextquellen* sein, d. h. ein Großteil des Nutzens der in diesem Kapitel dargestellten Verfahren hängt sehr stark von den verfügbaren Kontextquellen sowie den darin befindlichen Informationen ab. Gleiches gilt für die zur konkreten Implementation kontextsensitiver Lösungen aufgewendete *Kreativität*, die sich aufgrund der engeren Einbindung dieser Lösungen in realweltbezogene Vorgänge stärker auf die anwenderseitige Beurteilung von deren Nutzen auswirkt als dies bei anderen Paradigmen der Fall ist. Der von Anwendern wahrgenommene Nutzen kontextsensitiver Lösungen ist also sehr stark von der Art und Weise ihrer konkreten Implementation und nicht nur von den dahinter stehenden Konzepten, die ja in dieser Arbeit betrachtet wurden, abhängig (vgl. [Hul⁺97, S. 10]). Fernerhin ist insbesondere der von auf *Context Maps* basierenden kontextbezogenen Assistenzdiensten erbrachte Nutzen oft nicht direkt quantifizierbar, da hierdurch

	rel. DB (ERM)	OODB	XML	semWeb/RDF	Power (Topic Maps)	Context Cube	Henricksen	Context Toolkit	Context Fabric	Sentient Computing	Context Framework	Gaia	Schmidt	Context Maps
1 Allgemeine Verwendbarkeit	3,0	3,6	3,8	4,0	4,0	3,0	3,0	3,0	4,0	3,6	2,6	4,0	3,2	5,0
2 Multidimensionalität	2,0	5,0	5,0	5,0	5,0	5,0	2,0	5,0	4,0	5,0	3,0	4,5	4,5	5,0
3 Dynamik	1,7	4,3	2,7	2,7	2,3	3,7	2,7	3,0	4,0	4,0	3,3	4,7	3,7	5,0
4 Datenqualität	1,5	2,0	2,0	2,0	1,0	2,0	5,0	2,0	3,5	2,5	5,0	2,0	3,5	5,0
5 Datensicherheit	1,3	2,0	2,0	2,0	2,0	2,0	2,0	2,0	5,0	2,0	2,0	2,0	2,0	4,3
6 Integration heterogener CI	2,1	3,2	2,5	2,6	3,2	3,4	2,9	3,1	3,8	3,4	2,6	2,6	3,1	5,0
7 Endnutzertauglichkeit	2,7	1,0	2,3	2,3	3,7	2,3	1,3	3,7	2,3	1,7	1,7	2,0	2,3	4,3

(a) Durchschnitte aller Arbeiten

	rel. DB (ERM)	OODB	XML	semWeb/RDF	Power (Topic Maps)	Context Cube	Henricksen	Context Toolkit	Context Fabric	Sentient Computing	Context Framework	Gaia	Schmidt	Context Maps
1 Allgemeine Verwendbarkeit	1	1	1	1	1	1	2	1	1	1	1	1	1	5
2 Multidimensionalität	1	5	5	5	5	5	2	5	3	5	1	4	4	5
3 Dynamik	1	1	1	1	1	1	1	1	3	2	2	4	3	5
4 Datenqualität	1	2	2	2	1	2	5	2	2	2	5	2	2	5
5 Datensicherheit	1	2	2	2	2	2	2	2	5	2	2	2	2	4
6 Integration heterogener CI	1	2	1	1	1	1	1	1	1	1	1	1	1	5
7 Endnutzertauglichkeit	1	1	2	2	2	1	1	3	1	1	1	1	1	4

(b) Minima der Bewertungen aller Arbeiten

	rel. DB (ERM)	OODB	XML	semWeb/RDF	Power (Topic Maps)	Context Cube	Henricksen	Context Toolkit	Context Fabric	Sentient Computing	Context Framework	Gaia	Schmidt	Context Maps
1 Allgemeine Verwendbarkeit	1	1	3	3	3	2	1	1	3	1		3	1	5
2 Multidimensionalität		2	2	2	2	2		2	1	2	1	1	1	2
3 Dynamik		1	1	1	1	2		1	1	2		2	1	3
4 Datenqualität							2		1		2		1	2
5 Datensicherheit									3					1
6 Integration heterogener CI	1	2	2	2	5	4	2	2	5	4	1	2	1	11
7 Endnutzertauglichkeit					1									1

(c) Anzahl der mit voller Punktzahl (5 Punkte) bewerteten Anforderungen

Abbildung 120: Bewertung von Context Maps im Vergleich zu den verwandten Arbeiten

oft nicht die primären Aufgabenstellungen, sondern lediglich *sekundäre* Tätigkeiten wie deren Einbindung in die umgebenden Abläufe verbessert werden. Der von diesen Anwendungen realisierte Nutzen ist für einen Anwender zwar deutlich spürbar, aber nur schwer objektiv in monetären Größen quantifizierbar (vgl. [Hul⁺97, S. 10]).

So erleichtert der in Kap. 6.2.5 („Tourist Guide“, S. 193) dargestellte kontextsensitive Reiseführer den Bummel durch eine fremde Stadt zwar beträchtlich, indem er den Nutzer von vielen Nebentätigkeiten wie dem Bezahlen von Fahrkarten und dem Blättern im Reiseführer befreit, jedoch könnte ein Anwender prinzipiell auch mit einem gedruckten Reiseführer und genügend Kleingeld durch die fremde Stadt laufen. Ebenso kann ein Navigationssystem auch manuell, d. h. ohne einen expliziten Kontextbezug wie in Kap. 6.2.4 („Kontextsensitives Navigationssystem“, S. 186) beschrieben, programmiert werden. Der von einer kontextsensitiven Lösung bereitgestellte Nutzen findet sich eher in Form eines erhöhten *Komforts* für den Anwender, welcher jedoch nur schwer objektivierbar und darüber hinaus sehr stark von den individuellen Umständen und der schon angesprochenen Implementierungsqualität abhängig ist.

Daher wird auf den Versuch einer isolierten Kosten-Nutzen-Rechnung etwa hinsichtlich eines konkreten *return-on-investment* der in dieser Arbeit beschriebenen, auf Context Maps basierenden Lösungen verzichtet und deren Nutzen lediglich qualitativ dargestellt.

Dieses Manko kann zwar nicht endgültig entkräftet werden, in dieser Arbeit sind jedoch genügend Ansätze für ökonomische Potenziale der entwickelten Technologie sowie konkrete Anwendungsbeispiele, welche deren Nutzen zumindest umreißen, genannt worden. Als Beispiel sei wiederum der in Kap. 6.2.5 auf S. 193 beschriebene Touristenführer genannt, dessen Funktionalität ohne Context Maps weitaus aufwändiger und wartungsintensiver realisiert werden müsste. Gleiches gilt für sämtliche anderen in Kap. 6.2 genannten Anwendungen.

6.4.2 Schutz der Privatsphäre

BARKHUUS und DEY haben experimentell nachgewiesen und in [BaDe03b] beschrieben, dass durchschnittliche Nutzer durchaus bereit sind, einen Teil ihrer Kontrolle über elektronische Assistenzsysteme aufzugeben, wenn der dafür erreichte Gewinn an Funktionalität groß genug ist und sie kein allzu großes Mißbrauchspotential vermuten. Diese Erkenntnisse entsprechen dem in der Community verbreiteten optimistischen Ansatz, dass der Nutzen von kontextsensitiven Lösungen deren Mißbrauchsrisiko i. d. R. überwiegt und diese Lösungen daher sinnvoll sind (vgl. [Pove99, GrHo03]). Trotzdem stellt sich die Frage, wie der notwendige Schutz privater Informationen sichergestellt werden kann.

Mobile, pervasive und kontextsensitive Szenarien bestehen häufig aus in die Umwelt und Kleidung eingearbeiteten Sensoren und Endgeräten, die prinzipbedingt umfangreiche Informationen aus der Privatsphäre der Nutzer erfassen. In Kombination mit einer dezentralen, relativ unplanbaren, da spontan und größtenteils ohne menschliches Zutun vernetzten Topologie unterschiedlicher Endgeräte, Anwendungen und Informationsbestände verschiedener Nutzer können sich durchaus gravierende Problemstellungen im Bereich der Sicherheit der Privatsphäre ergeben (vgl. [Sloa92, Coy92, O.V.02, O.V.98b, O.V.00]). Diese Probleme sind erheblich und können zu einem *Showstopper* für kontextbezogene Technologien werden, d. h. den Einsatz von kontextbezogener Technologie regelrecht verhindern. In diesem Zusammenhang sei an das Scheitern der agentenbasierten Konzepte aus ähnlichen Gründen erinnert. HONG bezeich-

net den Schutz der Privatsphäre daher völlig zutreffend als „*tremendous source of valid criticism*“ für kontextsensitive Überlegungen (vgl. [Hon⁺03, S. 39]).

Wie bereits in Kap. 2.4.4 („Datensicherheit“, S. 33) dargestellt wurde, ist das Ziel dieser Arbeit nicht die möglichst vollständige Abschottung bzw. Unterdrückung der Nutzung privater Daten, sondern deren möglichst effektive Nutzung durch vom Anwender autorisierte Assistenzanwendungen. In diesem Zusammenhang erlauben Context Maps zwar nicht den gegen den Willen des Nutzers durchgeführten gewaltsamen Zugriff von nicht autorisierten Instanzen auf für sie eigentlich unerreichbare Informationen, tragen zu den genannten Problemen jedoch dahingehend bei, dass sie u. U. private Informationen aus verschiedenen Anwendungen zusammenführen und dadurch den Zugriff auf sie zumindest etwas vereinfachen.

Der Schutz der Privatsphäre kann in angemessener Form nur durch einen ganzen Katalog von Maßnahmen realisiert werden. Als selbstverständlich vorausgesetzt werden an dieser Stelle grundlegende datenschutztechnische Prinzipien wie eine Authentifizierung von Kommunikationspartnern sowie eine Verschlüsselung der ausgetauschten Inhalte. Ferner sieht das in dieser Arbeit entwickelte Kontextmodell explizite Zugriffsbeschränkungsmechanismen in Form eines auf der Rechtsauffassung des Bundesdatenschutzgesetzes basierenden Informationseigentümermodells sowie ein die Zugriffs- und die Verarbeitungsrechte auf die enthaltenen Informationen regelndes Informationszugriffsmodell vor, welches von HONG entwickelt und validiert wurde (vgl. [Hong04, Fein00]).

Eine weitere, allerdings architektonische und damit in dieser Arbeit nicht weiter vertiefte Möglichkeit zur Wahrung der Vertraulichkeit privater Informationen bei externen Instanzen, denen nicht vertraut werden kann, besteht darin, zu verhindern, dass die privaten Informationen die Domäne des jeweiligen Anwenders verlassen, was bedeutet, dass sämtliche diese Informationen nutzenden Anwendungen auf einem Rechner des Nutzers ausgeführt werden müssen, wobei auch die Ergebnisse dieser Operationen diesen Bereich nicht mehr verlassen dürfen, da sie u. U. Rückschlüsse auf die dabei verwendeten Nutzerinformationen zulassen. Dies bedeutet, dass sich eine Anwendung des Nutzers bei Bedarf sämtliche verfügbaren Informationen von außen beschaffen und die entsprechende Funktionalität komplett selbst erledigen muß (vgl. Abb. 121 auf der gegenüberliegenden Seite). Dieses Konzept wurde von MANIATIS in Form der *Mobile People Architecture*, welche einen *Personal Proxy* beinhaltet, der sämtliche privaten Daten verwaltet, realisiert (vgl. [Man⁺99]).

Durch diesen *Personal Proxy*-Ansatz können eine ganze Reihe von Problemen bei der Verarbeitung vertraulicher Informationen umgangen werden, jedoch ist er nicht für sämtliche Problemstellungen angemessen, da die dabei zu übertragenden Datenmengen zu groß oder rechtlich geschützt sein können, etwa bei der Darstellung des aktuellen Ortes auf einer hochauflösenden Landkarte, von der letztlich nur ein kleiner Ausschnitt dargestellt werden soll. In diesem Fall müssen private Daten also dennoch an einen externen Dienstleister gesendet werden, der eine entsprechend verarbeitete Antwort zurückliefert (vgl. Abb. 121(a) auf der gegenüberliegenden Seite). Falls dem externen Dienstleister nicht vertraut wird, können die betreffenden privaten Informationen dazu anonymisiert werden, was durch Nutzung von Pseudonymen erfolgen kann. Dabei müssen auch die Netzwerkadressen der verwendeten Endgeräte sowie die entsprechenden Zahlungsströme über Mittelsmänner anonymisiert werden, im Zweifelsfall

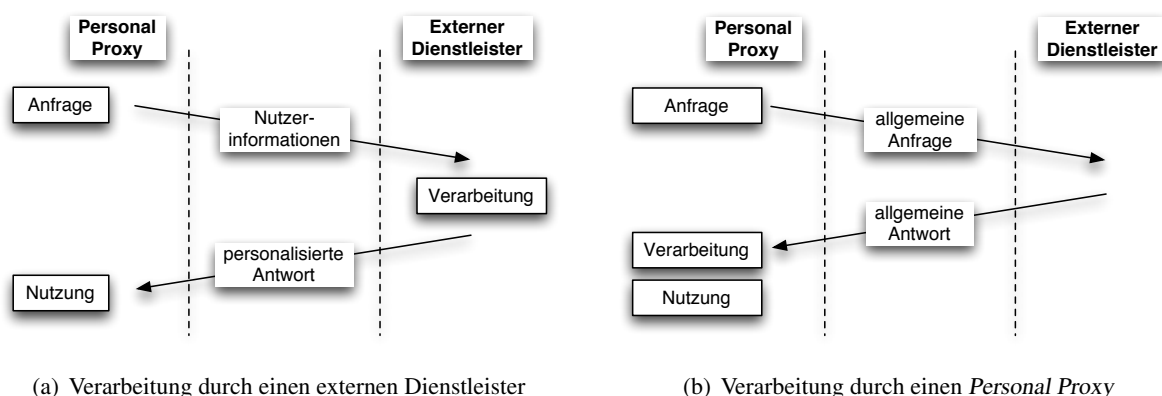


Abbildung 121: Formen der Verarbeitung von personenbezogenen Informationen innerhalb und außerhalb eines *Personal Proxy*

mittels einer von KESDOGAN und PFITZMANN beschriebenen mehrstufigen Kaskade von Anonymisierungsvorgängen (vgl. [Kes⁺96]).

Da die korrekte Realisierung der genannten Sicherheitsmechanismen transparent nur von quelloffenen Anwendungen bewiesen werden kann, spielen diese auch als *Open Source* bezeichneten Konzepte im Bereich der Verarbeitung privater Informationen eine wesentlich tragendere Rolle als in anderen Bereichen (vgl. [Han⁺02, Pere97, Pers99]).

Sämtliche genannten Maßnahmen regeln nur den *technischen* Aspekt des verantwortungsvollen Umgangs mit privaten Informationen. Hinsichtlich der *betriebswirtschaftlichen* Ebene formulierte WESTIN bereits 1967 die *fair information practices* als Richtlinien zum verantwortungsvollen Umgang mit persönlichen Informationen durch betriebswirtschaftliche Anbieter (vgl. [West67]). LANGHEINRICH hat diese Richtlinien auf kontextbezogene, ubiquitäre Szenarien angewendet, wobei jedoch anzumerken ist, dass diese aufgrund ihrer Komplexität und ihres methodischen Ansatzes in einer sinnvollen Weise nur auf Unternehmen und nicht etwa im privaten Bereich anwendbar sind (vgl. [Lang01], [Hong04, S. 2]). Die Idee einer unabhängigen datenschutztechnischen Zertifizierung von Anbietern hinsichtlich eines angemessenen und verantwortungsvollen *Sicherheitsmanagements* analog zum *Qualitäts-, Umwelt- oder Hygienemanagement* nach ISO 9000 ff, ISO 14000 ff oder HACCP steht nach wie vor im Raum, hat jedoch noch zu keinen konkreten Realisierungen geführt, welche für die nächste Zeit auch nicht absehbar sind, da der Bedarf hierfür anscheinend noch nicht groß genug ist.

Da hinter dem Konzept von kontextsensitiven Assistenten zumindest im Sinne dieser Arbeit letztlich überhaupt nicht die Absicht steht, einen *Big Brother*, der die Nutzer kontrolliert und herumkommandiert zu erschaffen (vgl. [Orwe49, O.V.98b]), sondern eher die Idee eines *Little Brother*, d. h. eines kleinen Helfers bzw. Werkzeugs, welches dem Nutzer Arbeit abnimmt und vollständig und ausschließlich vom Nutzer kontrolliert wird und damit nur das tut und weiß, was der Nutzer ihm erlaubt, sollte es letztlich möglich sein, unter Nutzung aller verfügbaren Sicherheitskonzepte hinreichend sichere kontextsensitive Assistenzsysteme zu schaffen. Die Aufgaben des speziell in dieser Arbeit dargestellten Informationsmanagementkonzeptes liegen dabei im Management von Informationen *des*, nicht *über* den Nutzer, d. h.

in der möglichst effektiven Sammlung und Integration aller sowieso bekannten, zur Zeit jedoch isoliert voneinander in heterogenen Datenfriedhöfen lagernden Kontextinformationen des Nutzers, die während seiner Tätigkeiten für ihn relevant sein könnten. Im Gegensatz zu den dafür heutzutage verwendeten *menschlichen* Aufgabenträgern kann der an dieser Stelle vorgestellte *maschinelle* Aufgabenträger so konstruiert werden, dass er immer motiviert, frei von persönlichen Interessen, über Korruption erhaben sowie lebenslang dem Nutzer bzw. dessen Unternehmen verpflichtet ist.

6.4.3 Zusätzlicher Modellierungsaufwand

Die Verwendung von *Context Maps* erfordert einen zusätzlichen Modellierungsaufwand für die separaten Kontextmodelle, welcher sich wie bereits in der Einordnung dieser Arbeit dargestellt erst bei relativ komplexen Szenarien, die erst durch die von *Context Maps* bereitgestellten Mechanismen sehr effektiv modelliert werden können, rentiert.

6.4.4 Integration in existierende Umgebungen

Es kann davon ausgegangen werden, dass es zu keinen grundlegenden Umstrukturierungen existierender Unternehmens- und IT-Landschaften nur zum Zweck der Einführung kontextsensitiver Konzepte kommen wird. Daher können auch nur sehr begrenzte Veränderungen an existierenden Anwendungssystemen vorausgesetzt werden, keinesfalls ein komplettes Re-Design, was wiederum zur Folge hat, dass die für umfassende kontextbezogene Assistenzfunktionen benötigten Informationen nicht aktiv von den betreffenden Quellanwendungen bereitgestellt werden.

Notwendig ist daher eine eigenständige, z. B. mittels des in dieser Arbeit vorgestellten Ansatzes realisierte Integration der heterogenen Informationsbestände. Ein entsprechender Kontextdienst muß daher selbst aktiv werden, um sich die benötigten Informationen zu beschaffen, auszuwerten und entsprechende Aktionen in anderen Anwendungen zu veranlassen. Dies kann z. B. im Rahmen einer *Enterprise Application Integration*-Strategie erfolgen (vgl. Kap. 4.2.2 auf S. 104).

Die erfolgreiche Realisierung kontextsensitiver Unternehmensprozesse hängt jedoch nicht nur von den durch *Context Maps* bereitgestellten flexiblen technischen Integrationsmöglichkeiten, sondern in wesentlichem Maß auch von den diesbezüglichen betriebswirtschaftlichen Führungsaufgaben ab. Dazu gehört die Einbettung der jeweiligen Infrastrukturkonzepte in die Unternehmensstrategie bzw. die einzelnen betrieblichen Informations- und Wissensmanagementstrategien, die Reorganisation bestehender Ablauf- und Informationssystemstrukturen hinsichtlich der neuen Möglichkeiten sowie auch eine Anpassung des Personalmanagements an die möglicherweise veränderten Anforderungen an die Mitarbeiter, etwa hinsichtlich einer stärkeren Überwachung oder anderer Entlohnungsparadigmen (vgl. Kap. 3.6.5 („Leistungsbezogener Vergütung“, S. 89)). Gleiches gilt für das inner- und zwischenbetriebliche Controlling, welches an die Art der durch *Context Maps* erzielbaren Mehrwerte und Effizienzsteigerungen angepasst werden muß.

6.4.5 Gesellschaftlicher Paradigmenwechsel

Da maschinelle Aufgabenträger beim kontextsensitiven Paradigma relativ stark in die Realwelt integriert sind und daher basierend auf und in Kombination mit der dadurch wesentlich umfassenderen Kenntnis über diese Realwelt eine stärkere Rolle in vielen Vorgängen in dieser Realwelt spielen können, bedingen sie durchaus signifikante gesellschaftliche Veränderungen.

Gegen Veränderungen gibt es jedoch immer einen allgemeinen Widerstand, der von den durch die Veränderung benachteiligten Gruppen aufgrund eines befürchteten Machtverlustes ausgeht sowie im konkreten Fall neben durchaus fundierten Argumenten vor allem mit einer teilweise irrationalen und durch mangelnde Sachkenntnis hervorgerufenen Angst vor Überwachung und Fremdsteuerung durch eine neue, nicht bis ins Detail verständliche Technik kombiniert und dadurch verstärkt wird. Diesen Argumenten kann dahingehend entgegengetreten werden, dass die schon seit Jahrtausenden, wohl schon seit dem Anbeginn der Menschheit bzw. sogar bereits in Tierrudeln auftretende Überwachung der Bevölkerung im allgemeinen noch nie auf besondere technische Hilfsmittel angewiesen war, sondern sich immer – durch ein entsprechendes politisches und gesellschaftliches Klima, in dem die Menschen anfangen, sich nicht mehr zu vertrauen hervorgerufen – vor allem im zwischenmenschlichen Bereich etabliert hat. In anderen Worten formuliert, bilden sich undemokratische Überwachungssysteme immer dann, wenn die entsprechenden wirtschaftlichen und politischen Rahmenbedingungen gegeben sind, und sie bilden sich in diesen Fällen zuverlässig und relativ unabhängig von den gegebenen technischen Mitteln, da das beste und i. d. R. auch einzige Mittel, wirklich in die Privatsphäre der zu observierenden Zielpersonen und an deren Gedankengut zu gelangen, nach wie vor menschliche Spione in Form von Freunden und Bekannten darstellen. Deshalb können Überwachungsprozesse mit Sicherheit nicht dadurch verhindert werden, dass entsprechende technische Mittel dafür fehlen.

Context Maps im speziellen können keine tragende Rolle in solchen Überwachungsszenarien spielen, da sie im Gegensatz zu bereits weit verbreiteten computerbasierten Spionagewerkzeugen keine Funktionen zum Eindringen in die Privatsphäre der zu observierenden Personen, sondern lediglich zur Kombination diverser heterogener Informationen bereitstellen. Überwachungsinformationen wurden jedoch schon immer miteinander kombiniert, weshalb in diesem Bereich bereits eine Fülle von Lösungen existiert. Der innovative Aspekt dieser Arbeit stellt die Anwendung dieses Prinzips im „legalen“ Kontextinformationsmanagement, d. h. zur Organisation von Informationen *innerhalb* der gegebenen Zugriffsrechte dar. Indem diese Form der „Überwachung“ zunehmend im privaten Umfeld eingesetzt wird, kann möglicherweise sogar das Bewusstsein einer breiten Bevölkerungsschicht für die tatsächlich durchgeführte Überwachung geschärft und die heute eher laxen Einstellung zu diesem Thema durch eine mehr auf *Informationshygiene*, d. h. den bewussteren Umgang mit den verursachten Datenspuren, basierende Herangehensweise ersetzt werden. Schließlich stellen etwa die bereits seit Jahrzehnten direkt in den privaten Wohnungen bzw. innerhalb der betrieblichen Büroumgebungen eingesetzten ISDN-Telefone für entsprechend interessierte Instanzen kostenlose „Wanzen“ dar, da sie basierend auf der Kombination verschiedener in der CCITT-Norm vorgesehener und damit von allen Telefonen unterstützter Funktionen jederzeit und unbemerkt für eine akustische Raumüberwachung verwendet werden können (vgl. [Wrig98, O.V.98a]). Gleiches gilt für die immer beliebter werdenden Mobiltelefone. Ebenfalls genannt werden sollen an dieser Stelle die in vielen Teilen der Welt bereits regulär und legal eingesetzten

kommerziellen Überwachungswerkzeuge für PC-Arbeitsplätze, die etwa Tastatureingaben aufzeichnen, den Bildschirminhalt oder Fotos des Benutzers mittels einer angeschlossenen Webcam in regelmäßigen Abständen aufzeichnen, sowie die Tatsache, dass der gesamte öffentliche Internet-, EMail-, Telefon- und Faxverkehr durch Systeme wie *Echelon* und andere Systeme, teilweise sogar durch entsprechende Gesetzesänderungen legitimiert, überwacht wird (vgl. [Camp00]).

Eine andere Ursache für eine (anfängliche) gesellschaftliche Skepsis gegenüber Context Maps stellen die dafür notwendigen Investitionskosten bei gleichzeitig bestehendem Investitionsrisiko dar, worunter im konkreten Fall Aufwendungen zur Veränderung der laufenden Geschäftsprozesse, d. h. Kosten für die neue Technik, Prozessumbau- und -stillstandszeiten, Weiterbildung von Anwendern sowie die Fehlerkosten in den ersten Produktgenerationen fallen. Diese Aufwendungen verursachen zum einen eine Einführung dieser Technologien in den bereits auf S. 21 dargestellten Wellen, zum anderen werden diese Technologien erst dann in breiter Front eingeführt, wenn deren Funktionalität und Praktikabilität durch diverse kommerzielle Anwendungsbeispiele öffentlichkeitswirksam bewiesen werden konnte.

6.4.6 Überformalisierung

Formalisierung ist eine notwendige Anforderung zur angemessenen Darstellung, Verarbeitung und zum Austausch von Kontextinformationen. Eine zu starke Formalisierung wirkt sich im allgemeinen jedoch einschränkend auf die Kreativität und Innovationsfähigkeit der die überformalisierte Domäne bearbeitenden Menschen aus, da hierdurch das Probieren neuer Möglichkeiten und damit das Finden entsprechenden Lösungen für die kontinuierlich auftretenden Problemstellungen erschwert sowie die daraufhin notwendigen Änderungen an Modellen und Abläufen nicht nur mit Implementations-, sondern auch mit Formalisierungsaufwand verbunden sind. Eine leichte Veränderbarkeit ist daher gerade bei kontextorientierten Verfahren aufgrund der Vielfältigkeit, Veränderlichkeit und der vielen Freiheitsgrade von realweltbezogenen Problemstellungen besonders wichtig, wobei an dieser Stelle auch auf die in Kap. 2.4.6 („Endnutzertauglichkeit“, S. 43) dargestellten Überlegungen verwiesen werden soll. Zudem ist die Lebenswelt der Anwender selbst überhaupt nicht vollständig strukturiert und damit prinzipbedingt auch nicht vollständig formalisierbar (vgl. [FuRo98]).

Teil der wirtschaftsinformatischen Gestaltungsaufgabe von kontextbezogenen betrieblichen Informationssystemen ist daher das Finden eines angemessenen Formalisierungskompromisses, der einerseits eine möglichst gute maschinelle Unterstützung von Realweltvorgängen und andererseits „*menschengerechte Informationssysteme für eine Kreativität fördernde Unternehmensorganisation*“ ermöglicht (vgl. [Fuc⁺99, S. 341]). Context Maps stellen einen solchen Formalisierungskompromiss dar, da sie die Welt zwar mittels durchaus formaler Ontologien strukturieren, andererseits jedoch auch explizit manuelle Veränderungsprozesse an diesen Strukturen vorsehen, etwa durch die in Kap. 2.4.6 („Endnutzertauglichkeit“, S. 43) beschriebene Endnutzertauglichkeit.

6.4.7 Skalierbarkeit des Modells

Aufgrund ihres umfassenden Ansatzes werden Context Maps im praktischen Einsatz sehr viele Informationen beinhalten, was zu umfangreichen und damit potentiell unübersichtlichen Kontextmodellen führen kann. Context Maps beinhalten jedoch gleich mehrere Verfahren zur Komplexitätsreduktion. Zum einen

stellen sie einen strukturierten Ansatz dar, welcher auch umfangreiche Modelle etwa durch mehrere Informationsebenen handhabbar macht (vgl. Kap. 5.5 („Schichtenarchitektur“, S. 162)). Weiterhin ist es möglich, Teile des Kontextmodells auf verschiedenen Abstraktionsstufen darzustellen und so nur die gerade benötigten Informationen in eine ansonsten grob vereinfachte Darstellung des Gesamtmodells einzublenden (vgl. Kap. 5.4.5.2 („Abstrakte Assoziationen“, S. 158)). Außerdem können Kontextinformationsbasen aus verschiedenen Context Maps zusammengesetzt werden, wodurch die Komplexität des Gesamtmodells in mehrere wiederverwendbare Teile untergliedert werden kann.

6.4.8 Performance

Da kontextsensitive Anwendungen häufig einen interaktiven Charakter haben, sollte die Beschaffung und Berücksichtigung von Kontextinformationen zeitnah, d. h. möglichst in Echtzeit erfolgen. Dies kann zu durchaus signifikanten Anforderungen an die Leistungsfähigkeit der diese Informationen verarbeitenden Anwendungssysteme führen, was besonders dann gilt, wenn eine detaillierte Historie eines ohnehin komplexen Kontextmodells zu speichern ist.

Zur Abschätzung der zu erwartenden Leistungsbeanspruchung sei an dieser Stelle erwähnt, dass existierende *Topic Map*-Engines laut ihren Herstellern durchaus in der Lage sind, auf üblicher Hardware Millionen von – allerdings statischen – Topics und Assoziationen performant zu verwalten.

Den mit Abstand größten Leistungsbedarf bei der Bewertung der in Kap. 6.1 („Realisierbarkeit“, S. 175) dargestellten Implementation einer Context-Map-Engine verursacht jedoch die Erzeugung, Kompilierung und Instanziierung von Aktiven Assoziationen. Da dieser Anwendungsbereich der Verarbeitung etwa von *Java Server Pages* ähnelt, welche bereits recht performant in betrieblichen Anwendungssystemen eingesetzt werden, können die dabei gewonnenen Erkenntnisse auf die Verarbeitung von Aktiven Assoziationen in Context Maps übertragen werden (vgl. [O.V.05b]). Bei einer geeigneten Implementation würde der Kompilierungsvorgang entsprechend nur einmalig bei der ersten Benutzung eines neuen Aktiven Assoziationstypes bzw. beim Import des entsprechenden Universums anfallen. Zur praxisnahen Abschätzung der Performanceanforderungen kontextbezogener Unternehmensanwendungen können weiterhin die Erfahrungen aus dem ODS-Bereich verwendet werden (vgl. [Inmo99]).

6.5 Zusammenfassung

Durch den Vergleich der Ergebnisse der Anforderungsanalyse von Context Maps und den verwandten Arbeiten konnte eindrucksvoll dargestellt werden, dass Context Maps verteilte und heterogene kontextbezogene Informationen in einer wesentlich geeigneteren Weise als bisherige Verfahren in einer integrierten Form darstellen können.

Die Frage der Realisierbarkeit von Context Maps wurde auf zwei Arten beantwortet. Zum einen wurde deren technische Durchführbarkeit durch Entwicklung einer funktionierenden Context-Map-Engine, welche die wichtigsten in diesem Zusammenhang verwendeten Konzepte funktionsfähig implementiert, bewiesen. Zum anderen wurde durch die Entwicklung diverser kontextsensitiver Anwendungen die *Funktionalität* von Context Maps demonstriert. Dabei konnte gezeigt werden, dass Context Maps durch ihren im Gegensatz zu bestehenden Kontextmodellen wesentlich umfassenderen Charakter nicht nur die Funktionalität von typischen Anwendungen in bereits hinreichend untersuchten Forschungsgebieten aus

dem Bereich der Kontextsensitivität weitreichender und komfortabler als vergleichbare Arbeiten gestalten können, sondern auch anwendungsübergreifende Prozesse durch Darstellung der diesbezüglichen Wirkungszusammenhänge der dabei auftretenden Kontextvariablen unterstützen können.

7 Zusammenfassung und Ausblick

7.1 Zusammenfassung

In dieser Arbeit wurde ein Verfahren zur Integration und Darstellung einer in verteilten und heterogenen Datenquellen befindlichen Menge von kontextbezogenen Informationen als Teil einer Managementstrategie zur systematischen und möglichst vollständigen Beschaffung von kontextbezogenen Informationen im Rahmen kontextsensitiver Geschäftsprozesse entwickelt.

Ausgangspunkt dieser Arbeit ist die Erkenntnis, dass umfassende Sichtweisen auf den Kontext einer Anwendung eine technisch wie betriebswirtschaftlich sinnvolle und daher anzustrebende Basis zur Entwicklung neuer, adaptiver Anwendungen darstellen, denn Kontextsensitivität stellt eine fundamentale Entwicklungsrichtung heutiger Informations- und Kommunikationssysteme sowie der damit realisierten Geschäftsprozesse dar, welche wiederum viele weitere Entwicklungen auf technischem, wirtschaftlichem sowie insbesondere auch wirtschaftsinformatischem Gebiet positiv beeinflusst. Dabei wurde festgestellt, dass die für wirklich *umfassende*, d. h. sämtliche bekannten Informationen über Realweltobjekte sowie sämtliche für eine bestimmte Anwendung relevanten Aspekte beinhaltende Sichtweisen benötigten Kontextinformationen bei den in der nahen Zukunft zu erwartenden Szenarien unvermeidbar in verschiedenen Datenquellen entstehen. Die Antwort auf den daraus resultierenden Integrationsbedarf stellt ein umfassendes, sämtliche bekannten Informationen über die modellierte Realweltdomäne beinhaltendes Kontextmodell dar, dessen einzelne Bestandteile mit Datensätzen in externen Datenquellen assoziiert sind und dadurch stets auf den aktuellen Stand gebracht werden können. Die diesbezüglichen Anforderungen wurden in Form von 7 Anforderungskomplexen definiert, welche sowohl zur Bewertung der bereits existierenden Lösungen auf diesem Gebiet als auch als Gradmesser für den Zielerreichungsgrad der eigenen Arbeit verwendet wurden.

Eine Analyse dieser verwandten Arbeiten offenbarte das Fehlen einer sämtliche Anforderungen befriedigenden Lösung, was die Entwicklung eines eigenen Integrations- und Darstellungsmodells für verteilte und heterogene Kontextinformationen notwendig macht. Dieses Modell stellt neben der strukturierten Beschreibung der betreffenden Problemdomäne mittels diverser Beschreibungsmodelle sowie einer Skizzierung der daraus resultierenden wirtschaftlichen und wirtschaftsinformatischen Potenziale den wissenschaftlichen Beitrag dieser Arbeit dar.

Das in dieser Arbeit entwickelte Darstellungsmodell für aus verteilten und heterogenen Datenquellen stammende, zu umfassenden Sichtweisen integrierte kontextbezogene Informationen basiert auf dem Prinzip des *Semantic Web* und insbesondere auf dem bekannten *Topic Map*-Standard, welcher um Konzepte zur inhaltlichen Darstellung von externen Informationen, zur expliziten Modellierung des Beschaffungsvorganges von Informationen, zur Darstellung der Historie und Datenqualität, der Zugriffsrechte und Dynamik der einzelnen Modellelemente sowie um Konzepte zur Integration methodischer und diverser modularer Elemente, um mit dem Modell in einer ergonomischen und flexiblen Weise umgehen zu können, erweitert wird.

Das gesamte Modell basiert auf einem expliziten, theoretisch fundierten Informationsintegrations- sowie einem Kontextdarstellungsmodell und definiert auf insgesamt 5 Schichten sämtliche zum Zugriff, der Integration sowie semantisch aussagefähigen Darstellung benötigten Konzepte. Das *Informations-*

integrationsmodell besteht dabei aus einem *Zugriffsmodell*, welches einen einheitlichen Zugriff auf Datensätze in verteilten und heterogenen Kontextquellen bereitstellt, einem *Integrationsmodell*, welches die durch das Zugriffsmodell adressierten externen Kontextvariablen zu expliziten Attributen von Realweltobjektmodellen fusioniert, einem *Verbindungsmodell*, welches die Zusammenhänge zwischen den Realweltobjektrepräsentationen selbst darstellt sowie einem *Verteilungsmodell*, welches die Verteilung der so integrierten Kontextinformationsbasis auf diverse Endgeräte transparent modelliert. Das *Kontextdarstellungsmodell* definiert in Form von *Universen* diverse ontologische Konzepte zur Formalisierung des zur Verarbeitung kontextbezogener Informationen notwendigen Domänenwissens. Dazu zählen *Ontologien*, welche die Typen der dargestellten Realweltobjekte in einer Vererbungshierarchie definieren sowie *Szenen*, welche Schablonen für typische im Kontext auftretende Situationen darstellen und das in dieser Situation geltende *Rollenmodell* für Realweltobjekte, die für die einzelnen Rollen geltenden Bedingungen sowie ggf. auch die von diesen Rollen auszuführenden Aktionen definieren.

Validiert wurde diese Arbeit in mehreren Hinsichten. Die *technische Durchführbarkeit* des Verfahrens wurde durch Entwicklung einer funktionierenden Context-Map-Engine demonstriert. Die *funktionalen Mehrwerte* wurden durch die Modellierung typischer kontextsensitiver Anwendungen demonstriert, welche aufgrund des durch Context Maps verfügbaren umfassenderen Kontextwissens wesentlich weitergehende bzw. komfortablere Funktionalitäten als vergleichbare verwandte Arbeiten realisieren können. In diesem Zusammenhang wurde ebenfalls die in dieser Arbeit postulierte Kombination von Informationen aus verschiedenen Anwendungen demonstriert.

Der *konzeptionelle Mehrwert* des in dieser Arbeit entwickelten Darstellungsmetamodells wurde durch eine Anforderungsanalyse validiert. Dabei wurde die Funktionalität sämtlicher betrachteter Verfahren hinsichtlich der in verschiedenen Anforderungskomplexen gebündelten und durch einzelne Kriterien objektiv vergleichbar formulierten Anforderungen bewertet. Auch hierbei konnte der Mehrwert von Context Maps gegenüber existierenden Verfahren deutlich indiziert werden. Der *wirtschaftliche Nutzen* der entwickelten Verfahren in den wichtigsten Anwendungsbereichen wurde ebenfalls umrissen.

Zusammenfassend lässt sich sagen, dass Context Maps eine Methode darstellen, um den sowohl innerhalb von Unternehmen als auch auf Seiten von deren Kunden und Zulieferern, dabei sowohl bei menschlichen als auch den von ihnen eingesetzten maschinellen Aufgabenträgern entstehenden permanenten Bedarf nach umfassenden Informationen über die aktuellen Umgebungsverhältnisse in angemessener Weise zu befriedigen. Dadurch können die betrieblichen Aufgabenträger in die Lage versetzt werden, angemessener mit den heterogenen und vielfältigen situationsbezogenen Problemstellungen im Unternehmensalltag umgehen zu können, etwa im Rahmen kontextsensitiver Unternehmensprozesse. Damit stellen *Context Maps* einen wichtigen Schritt auf dem Weg zur realen Utopie der Wirtschaftsinformatik, der sinnhaften Vollautomation des betrieblichen Geschehens, dar (vgl. [Mert95]).

7.2 Ausblick

Anknüpfungspunkte für weitere Forschungsarbeiten finden sich aufgrund der relativen Neuheit des Konzeptes an vielen Stellen. In wirtschaftsinformatischer Hinsicht gilt es, das in dieser Arbeit entwickelte Verfahren in umfassendere wirtschaftsinformatische Vorgehens- und informationsmodelle zu integrieren.

Konzeptionell ist eine Transformation der Informationsbasis in verschiedene anwendungs- bzw. problemspezifische Ontologien derzeit beschrieben, wird jedoch nicht weitergehend unterstützt. Entsprechende Weiterentwicklungen zur Automatisierung dieser *Universentransformation* beinhalten die Ableitung bzw. zumindest Modellierung interontologischer Beziehungsinformationen wie Synonyme, Hypo- und Hypernyme zur Zuordnung von Konzepten einer Ontologie zu den Konzepten in anderen Ontologien. Hier kann auf die Arbeiten von MENA verwiesen werden (vgl. [Men⁺98]).

Auch wurde in dieser Arbeit die Notwendigkeit ausgereifter Sicherheitskonzepte erkannt und ansatzweise in das Modell integriert. Wirklich praxistaugliche Konzepte, welche einen angemessenen Zugang zu personenbezogenen Informationen ohne zu viele Rückfragen bei deren Eigentümer ohne zu laxen Zugriffsrichtlinien ermöglichen, werden jedoch auf die Auswertung weiterer Kontextinformationen angewiesen sein, existieren derzeit aber noch nicht. Die heutzutage verwendeten Ansätze, Akteure in feste Gruppen einzuteilen und die Zugriffsrechte für diese Gruppen festzulegen, stellen einen Anfang, aber noch nicht das wirklich praxistaugliche Ende dieser Entwicklung dar. Weitere Informationen über den *Sicherheitskontext* einer Anfrage, etwa die Identifikation der anfragenden Anwendungen, deren Nutzer, die Verfügbarkeit ausreichend verschlüsselter Kommunikationswege sowie die mögliche Verwendung der Informationen sind notwendig, um auch in fortgeschrittenen Anwendungsfeldern *angemessene* Entscheidungen über die Freigabe ggf. auch situationsabhängig modifizierter, etwa anonymisierter oder verrauschter personenbezogener Informationen treffen zu können.

Auch könnten durch die in dieser Arbeit entwickelte Verschmelzung von vernetzten Daten und Methoden weitere, neben den eben bereits dargestellten sicherheitsbezogenen auch anwendungsbezogene Aspekte in die Kontextdatenstrukturen Einzug finden, welche in dieser Arbeit noch ausgeklammert werden mussten. Dazu müssen Anwendungsaspekte von durch aktive semantische Netze modellierten Informationen in weiteren Anwendungsdomänen identifiziert und untersucht werden.

Obwohl sicherlich nur teilautomatisiert möglich, stellt auch die Komposition der entsprechenden Kontextmodelle basierend auf einer automatischen Zuordnung von benötigten mit den von den einzelnen Kontextquellen bereitgestellten Kontextvariablen eine interessante Entwicklungsrichtung dar. Zu nennen sind hier die Arbeiten von HONG und COHEN (vgl. [Hong01, Coh⁺01]).

Auf der technischen Seite sind vor allem performante Aspekte zu nennen, die sowohl konzeptueller als auch implementationstechnischer Natur sind und in weiterführenden Arbeiten untersucht werden müssen. Die Erfahrungen mit der in dieser Arbeit als *proof-of-concept* entwickelten und daher keine speziellen Optimierungen beinhaltenden Context-Map-Engine, welche die Codesegmente im Datenmodell zur Laufzeit dynamisch kompiliert und instanziiert, verursachen selbst auf leistungsfähigen stationären Rechnersystemen wahrnehmbare Verarbeitungszeiten. An dieser Stelle wären Erfahrungen aus anderen Lösungen, welche zur Laufzeit Java-Code erzeugen, kompilieren und instanziierten, etwa bei der Verarbeitung von *Java Server Pages*, anzuwenden.

Ähnliche Überlegungen können zum Speicherplatzverbrauch angestellt werden, was sich vor allem auf Verfahren zur Kompression und Bereinigung der Historie von Kontextvariablen in Abhängigkeit der verfügbaren Ressourcen und dem von Anwendungen benötigten Umfang beschränkt, da das Kontextmodell selbst bereits weitgehend von Redundanz befreit ist.

Auch verwendungsseitig, d. h. auf wirtschaftlichem und wirtschaftsinformatischem Gebiet konnten die möglichen Einsatzgebiete von umfassenden Kontextinformationen in betrieblichen und technischen Anwendungssystemen aufgrund des fächerübergreifenden Charakters dieser Arbeit nicht abschließend dargestellt werden. Weiterführende Arbeiten müssen den Einfluß von umfassenden Kontextinformationen auf betriebswirtschaftliche Prozesse bzw. die entsprechenden Prozessmodelle in einer formalisierteren Form beschreiben und quantifizieren als dies in dieser Arbeit möglich war. Gleichfalls relevant sind Erhebungen über Erfahrungen mit in der Praxis tatsächlich eingesetzten kontextsensitiven Anwendungssystemen, auf die zum gegebenen Zeitpunkt nicht zurückgegriffen werden konnte, da die Verwendung von kontextsensitiven Unternehmensanwendungen gerade erst im Entstehen begriffen ist und noch lange nicht kostendeckend eingesetzt wird. Aufgrund des innovativen Charakters des gesamten Ansatzes, welcher bei richtiger Realisierung einen erheblichen Wettbewerbsvorteil darstellt, gehen die betreffenden Unternehmen zudem sehr restriktiv mit ihren kontextbezogenen Erfahrungen um.

Abbildungsverzeichnis

1	Umweltbezug von menschlichen Anwendern und deren Werkzeugen	2
2	Funktionale Erweiterungen von Computersystemen	2
3	Das Verteilungsproblem bei heterogenen Kontextquellen	8
4	Darstellung des Integrationsproblems	9
5	Beispiele für semantische Zusammenhänge zwischen realer Welt und deren Repräsentation .	14
6	Grundmodell der Unternehmung nach GROCHLA ([Groc75])	15
7	Das betriebliche Objektsystem nach FERSTL und SINZ	16
8	Integration heterogener Informationsmengen in umfassende Sichtweisen	18
9	Schematischer Aufbau von Kontextquellen	20
10	Entwicklungswellen von kontextbezogener Technologie	21
11	Heterogene Kontextquellen	23
12	Verteilungsmodell für Kontextinformationen	24
13	Vernetzung der Dienste von Endgeräten und Infrastruktur	27
14	Adressierungsarten	29
15	Variable Repräsentationen einer Kontextvariablen für verschiedene Zielgruppen (vgl. [Hon ⁺ 03, S. 52]	35
16	Realweltobjektbezogene Strukturierung von Kontextvariablen	36
17	Doppelte Datenhaltung	38
18	Trennung einer kontextsensitiven Anwendung in einen Business- und Kontextteil	40
19	Abgegrenzte, subjektive Sichtweisen auf die Datenbasis	46
20	Wiederverwendbare problemspezifische Sichtweisen	46
21	Genauigkeit der Bestimmung der räumlichen Orientierung von Endgeräten i. A. von Anzahl und Abstand der dazu verwendeten Sensoren ([Har ⁺ 01, S. 3]	50
22	Aufgabenkontinuum kontextbezogener Anwendungssysteme	51
23	Wearable Computing im betrieblichen Außendienst	54
24	Augmented Reality bei der Fahrzeugwartung	56
25	Entwicklung der kumulierten Verkaufszahlen von Rechnersystemen der drei Computing-Paradigmen (vgl. [Weis91]	57
26	Arten von Eingabekanälen in Computeranwendungen	79
27	Eingabekanäle in Computeranwendungen im kontextsensitiven und nicht-kontextsensitiven Fall	80
28	Klassifikation der durch Automatisierung zu unterstützenden Aufgabenbereiche hinsichtlich ihres Innovationsgrades	82
29	Zusammenhang von Pro- und Reaktivität	83
30	Personalisierte Verwaltung von Einkaufslisten im <i>Future Store</i> der METRO GROUP	88
31	Kontextebenen	98
32	3-Ebenen-Modell des <i>Context Toolkit</i>	99
33	Organisation von Kontextvariablen im <i>TEA</i> -Modell	99
34	Aufbau einer Active Map	100

35	Ausschnitt aus einem Context Schema für den Aufenthalt eines U-Bootes im Hafen	102
36	Datenintegration	103
37	Typische Topologie eines EAI-Systems	105
38	Beispiel für ein typisches in einem ODS verwendetes Datenschema (vgl. [Inmo99, S. 86]) . .	106
39	3-Ebenen-Struktur des <i>Semantic Web</i>	107
40	3 Ebenen der Kontextdatenverarbeitung	108
41	Interpretation von Kontextinformationen durch externe Agenten	108
42	Kapselung der Interpreter in Objektstrukturen	109
43	Integration von Interpretern in die Datenstruktur mittels <i>aktiver Assoziationen</i>	110
44	Automatische Generierung eines passenden Transformationspfades	111
45	Ebenenmodell zur Integration verteilter Kontextinformationen	112
46	Merging von Kontextontologien durch POWER	116
47	Context Cubes	118
48	Kontextmodell nach HENRICKSEN	120
49	Schichtenarchitektur des Kontextmodells des <i>Context Toolkit</i>	122
50	Vollautomatische Bestimmung von Pfaden am Beispiel des Findens in der Nähe befindlicher Kinos	123
51	Struktur des <i>Context Fabric</i>	123
52	<i>Context Shadow</i> von JONSSON	124
53	Aufbau eines <i>Sentient Object</i>	125
54	Aufbau einer aus <i>Sentient Objects</i> aufgebauten kontextsensitiven Anwendung	126
55	Integrationsprinzip verteilter Kontextinformationen beim <i>Context Framework</i> (vgl. [Kadn04a, S. 48])	127
56	Integrationsstruktur des <i>Gaia-Frameworks</i>	129
57	Überblick über die Bewertung der in dieser Arbeit diskutierten Arbeiten zur Integration und Darstellung verteilter und heterogener Kontextinformationen	132
58	Durchschnittliche Bewertung der Anforderungskomplexe	133
59	Minima bei der Bewertung der Anforderungskomplexe	134
60	Visualisierung des Zugriffsmodells	136
61	Modell zur Integration isolierter Kontextvariablen in einen Attributwert eines Realweltobjektes	137
62	Schematische Darstellung des Verbindungsmodells	137
63	Zugriffsmöglichkeiten auf Kontextinformationen	138
64	Auf diverse Geräte und Maps verteilte, aber miteinander verbundene Datenhaltung	138
65	Stufenweise anwendungsspezifische Verdichtung der Datenbasis	139
66	Gesamtes theoretisches Integrationsmodell	139
67	Kombination von elektronischen mit Realweltuniversen zu elektronischen Realweltmodellen	141
68	Anzahl der mit voller Punktzahl (5 Punkte) erfüllten Anforderungen	145
69	Notation für Topics	147
70	XML-Notation für Topics	147
71	XML-Notation für Namen von Konzepten	148

72	XML-Notation für <i>Subject Indicators</i>	149
73	XML-Notation für Occurrences	151
74	XML-Notation für <i>Subject Indicators</i>	152
75	XML-Notation für eine unbekannte Kontextvariable	152
76	Orthogonaler Zusammenhang zwischen Realweltmodellen und Beschaffungsmaps	153
77	Beispiel für eine Beschaffungsmap	153
78	XML-Notation für die Historie von Attributen	154
79	XML-Notation für Qualitätsdeskriptoren von Attributen	156
80	XML-Notation für Qualitätsdeskriptoren von Attributen	158
81	Graphische Notation für normale Assoziationen	158
82	XML-Notation für eine normale Assoziation	159
83	Darstellungsarten einer Abstrakten Assoziation	160
84	XML-Notation für eine Abstrakte Assoziation	160
85	XML-Notation für eine Aktive Assoziation	162
86	XML-Notation für den Scope, welcher den englischen Namen der Stadt München beschreibt	163
87	Schematische Darstellung des 5-Ebenen-Modell von <i>Context Maps</i>	164
88	Graphische Notation für <i>Ontology-Maps</i>	165
89	XML-Notation für eine <i>Ontology-Map</i>	166
90	Graphische Notation für <i>Template-Maps</i>	167
91	Kombination der separat modellierten Modellelemente zu pragmatischen Realweltmodellen .	168
92	Graphische Notation für <i>Instance-Maps</i>	168
93	Beispiel für ein Kontextquellentypmodell	170
94	Transformation einer <i>Context Map</i> in verschiedene Ontologien	171
95	Realweltuniversum des Nutzerprofils im Ausgangszustand des Szenarios	177
96	Beschaffungsmaps für Attribute des Topics <i>Nutzer</i>	177
97	Instanz des Nutzerprofils im dargestellten Szenario	178
98	Realweltuniversum des Messenger-VUI	180
99	Anfrageschablone für die Benutzung des Kontextmodells des Messenger-VUI	180
100	Ergebnis	181
101	Universum der Restaurant-Finder-Anwendung	182
102	Schrittweiser Aufbau einer Modellinstanz der Restaurant-Finder-Anwendung	182
103	Bestimmung der Yellow Pages von Paris	183
104	Bestimmung aller Restaurants in Paris	184
105	Herausfiltern der zu weit entfernten Restaurants	184
106	Entfernung der für die Nutzeranforderungen unpassenden Restaurants	185
107	Endgültiger Zustand der Kontextmodellinstanz des Restaurant-Finders	186
108	Manuelle Selektion des Fahrtziels im Text einer EMail	187
109	Volltextsuche nach "kevin" im Datenbestand des Nutzers	188
110	Liste der automatisch bestimmten Fahrtziele des Nutzers	188
111	Gesamte Darstellung des SCANav-Realweltuniversums	189

112	High-level-Sichtweise des Kontextmodells des Navigationssystems	190
113	Low-Level-Sichtweise auf die Bestimmung des Fahrers	190
114	Bestimmung der Fahrtziele des Fahrers	191
115	Low-level-Map der Assoziation <i>Nutzer hat Verabredung</i>	191
116	Von SCANav per Text-Mining bestimmte Informationen in einer EMail	192
117	Kontextmodell mit aufgeklappter Assoziation <i>Verabredung hat Fahrtziel</i>	192
118	Low-level-Map der Abstrakten Assoziation <i>Kontakt an Ort</i>	193
119	Template des <i>Tourist Guide</i>	195
120	Bewertung von Context Maps im Vergleich zu den verwandten Arbeiten	198
121	Formen der Verarbeitung von personenbezogenen Informationen innerhalb und außerhalb eines <i>Personal Proxy</i>	201
122	DTD der gesamten Context Map	LXVII
123	DTD eines Realweltuniversums	LXVII
124	DTD einer <i>Instance-Map</i>	LXVIII
125	DTD von Kontextquellentypen	LXVIII
126	DTD einer Map	LXVIII
127	DTD eines Topics	LXIX
128	DTD eines Attributes	LXIX
129	DTD einer <i>Normalen Assoziation</i>	LXIX
130	DTD einer <i>Abstrakten Assoziation</i>	LXX
131	DTD einer <i>Aktiven Assoziation</i>	LXX
132	Format der erweiterten URI-Syntax	LXXI
133	Beispiele für die direkte Adressierung eines Datensatzes mittel der erweiterten URI- Syntax	LXXI

Tabellenverzeichnis

1	Übersicht über die grundlegenden Kontextdimensionen	13
2	Beispiele für Kontextvariablen (vgl. [Mei ⁺ 04, S. 22])	13
3	Arten von Beziehungen zwischen Diskurs- und Umweltobjekttypen	17
4	Schema zur Bewertung von Lösungsansätzen hinsichtlich der Integration von heterogenen Kontextinformationen	26
5	Qualitätsmängel von Kontextdaten	32
6	Qualitätsattribute für Kontextvariablen (vgl. [Hen ⁺ 02, S. 178])	33
7	Anforderungen an Integrationsmodelle für kontextbezogene Informationen	48
8	Konfigurationsdimensionen	70
9	Arten und Handhabungen typischer kontextbezogen erkennbarer Fehlerklassen	71
10	Kommunikationskontexte	76
11	Phasen von Dienstleistungen	93
12	Für das Matching beim Restaurantsuchprozeß relevante Kontextinformationen	94
13	Assoziationsaspekte im Model von HENRICKSEN	120
14	Elemente des Kontextmodells des <i>Context Toolkit</i>	121
15	Arten von <i>Sentient Objects</i>	126
16	Von Gaia ausgewertete Kontextinformationen	128
17	Beispiele für verschiedene bei der Benutzung des Kontextmodells von <i>Gaia</i> verwendete Aussagen in Prädikatenlogik	129
18	Tupel zur Beschreibung einer Kontextvariablen im Kontextmodell von <i>Schmidt</i>	130
19	Zum Zugriff auf externe Kontextvariablen benötigte Informationsarten	136
20	Definitionsarten für Subject Identity	149
21	Schemata von im <i>Context Map</i> -Standard verwendeten URI	150
22	Attributelemente	151
23	Beispiele für nutzerbezogene Qualitätsdeskriptoren	155
24	Arten von Datenschutzdeskriptoren	157
25	Von einem kontextsensitiven Navigationssystem verwendete Kontextquellen	190
26	Vom <i>Tourist Guide</i> verwendete Kontextquellen	194
27	Bewertung des Entity-Relationship-Modells hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	XXXVII
28	Bewertung des objektorientierten Datenmodells hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	XL
29	Bewertung der Extensible Markup Language hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	XLII
30	Bewertung des Semantic Web hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	XLIV
31	Bewertung des <i>Topic Map</i> -Ansatzes hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	XLVI

32	Bewertung des <i>Context Cube</i> -Ansatzes hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	XLVIII
33	Bewertung des Ansatzes von HENRICKSEN hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	L
34	Bewertung des Datenmodells des <i>Context Toolkit</i> hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	LII
35	Bewertung des <i>Context Fabric</i> hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	LIV
36	Bewertung von <i>Sentient Computing</i> hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	LVI
37	Bewertung des Kontextmodells des <i>Context Frameworks</i> hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	LVIII
38	Bewertung von <i>Gaja</i> hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	LX
39	Bewertung des tupelraumbasierten Kontextmodells von SCHMIDT hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	LXII
40	Bewertung des Kontextmodells des Owl Context Service hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen	LXIV

Abkürzungsverzeichnis

API	Application Programming Interface
BDE	Betriebsdatenerfassung
CBR	Case-Based Reasoning
CC	Context Cube
CEO	Chief Executive Officer
CCDM	Context Cube Datenmodell
CIM	Computer Integrated Manufacturing
CIP	Context Information Provider
CIS	Context Information System, Kontextinformationssystem
DSS	Decision Support System
EAI	Enterprise Application Integration
EJB	Enterprise Java Beans
ERM	Entity-Relationship-Modell
EUP	End User Programming
GPS	Global Positioning System
HACCP	Hazard Analysis and Critical Control Point
HCI	Human Computer Interface
HVAC	Heating, Ventilation, Air Condition, Cooling
IuK	Information und Kommunikation
IR	Information Retrieval
ODS	Operational Data Store
OLAP	Online Analytical Processing
OODB	Objektorientierte Datenbank
OODM	Objektorientiertes Datenmodell
PDA	Personal Digital Assistant
PPS	Produktionsplanungs- und Steuerungssystem
PSI	Published Subject Indicator
RDB	relationale Datenbank
RDBS	relationales Datenbanksystem
RDF	Resource Description Framework
RDM	relationales Datenmodell
SQL	Structured Query Language
SI	Subject Indicator
TCO	Total Cost of Ownership
TM	Topic Map
UDM	Unternehmensweites Datenmodell
UDDI	Universal Description, Discovery and Integration
URI	Uniform Resource Identifier
VUI	Voice User Interface

W3C	World Wide Web Consortium
WLAN	Wireless LAN
WWW	World Wide Web
XML	Extensible Markup Language

Literaturverzeichnis

- [Abe⁺98] ABECKER, A.; BERNARDI, A.; HINKELMANN, K.; KÜHN, O.; SINTEK, M.: *Toward a Technology for Organisational Memories*. In: IEEE Intelligent Systems & Their Application, (1998), May/June
- [Abe⁺00] ABECKER, A.; BERNARDI, A.; DENGEL, A.; VAN ELST, L.; MALBURG, M.; SINTEK, M.; TABOR, S.; WEIGEL, A.; WENZEL, C.: *FRODO: A Framework for Distributed Organizational Memories*. In: *Project Proposal FRODO*. DFKI GmbH, Kaiserslautern, 2000
- [Abo⁺97] ABOWD, G. D.; ATKESON, C. G.; HONG, J.; LONG, S.; KOOPER, R.; PINKERTON, M.: *Cyberguide: A mobile context-aware tour guide*. In: Wireless Networks, 38 (1997) 5, S. 421–433
- [Adam79] ADAMS, D.: *The Hitchhikers Guide to the Galaxy*. Pan Macmillan Publishers Ltd., London, 1979
- [AgRo97] AGRE, P. E.; ROTENBERG, M.: *Technology and Privacy: The New Landscape*. MIT Press, Cambridge, USA, 1997
- [Ail⁺02] AILISTO, H.; ALAHUHTA, P.; HAATAJA, V.; KYLLÖNEN, V.; LINDHOLM, M.: *Structuring Context Aware Applications: Five-Layer Model and Example Case*. In: *Workshop for concepts and models for ubiquitous computing*, 2002
- [Alla97] ALLAN, G.: *What is Configuration Management?* In: Logistics Spectrum, 31 (1997) 1, S. 15–18
- [Ash⁺01] ASHBROOK, D.; AUXIER, J.; GANDY, M.; STARNER, T.: *Experiments in Interaction Between Wearable and Environmental Infrastructure Using the Gesture Pendant*. In: *Human Computer Interaction International (HCII), New Orleans*, 2001
- [AsSt02] ASHBROOK, D.; STARNER, T.: *Learning Significant Locations and Predicting User Movement with GPS*. In: *Int. Symp. on Wearable Computers, Seattle*, 2002, S. 101–108
- [Atk⁺95] ATKINSON, M.; BANCILHON, F.; DEWITT, D.; DITTRICH, K.; MAIER, D.; ZDONIK, S.: *The Object-Oriented Database System Manifesto*.
<http://www-2.cs.cmu.edu/People/clamen/OODBMS/Manifesto/htManifesto/Manifesto.html>, 1995
- [AuFr01] AUTH, G.; FRIE, T.: *Kopplung operativer (horizontaler) Applikationen mit dem Data Warehouse. Arbeitsbericht des Kompetenzzentrums "Data Warehousing 2"*. Universität St. Gallen, Forschungsbericht, 2001
- [Azum01] AZUMA, R. T.: *Augmented Reality: Approaches and Technical Challenges*. In: BARFIELD, W. (Hrsg.); CAUDELL, T. (Hrsg.): *Fundamentals of Wearable Computers and Augmented Reality*. Lawrence Erlbaum Associates, 2001, Kapitel 2, S. 27–63

- [Bac⁺02] BACK, A.; BENDEL, O.; STOLLER-SCHAI, D.: *E-Learning: Ein Wörterbuch*. Kappelrodeck, 2002
- [BaDe03a] BARKHUUS, L.; DEY, A.: *Location-based services for mobile telephony: a study of users' privacy concerns*. In: *Proceedings of INTERACT 2003, 9th IFIP TC13 International Conference on Human-Computer Interaction*, 2003
- [BaDe03b] BARKHUUS, L.; DEY, A.: *Is Context-Aware Computing Taking Control Away from the User? Three Levels of Interactivity Examined*. In: *Proc. of the 5th Annual Conference on Ubiquitous Computing (UbiComp 2003)*, 2003
- [BaKi01] BARTON, J.; KINDBERG, T.: *The Cooltown User Experience*. HP Laboratories Palo Alto, Internet and Mobile Systems Laboratory, Forschungsbericht, 2001
- [Bang95] BANGEMANN, A.: *Europas Weg in die Informationsgesellschaft*. In: *Informatik Spektrum*, 18 (1995) 1, S. 1–3
- [BaRi01] BAEZA-YATES, R.; RIBEIRO-NETO, B.: *Modern Information Retrieval*. ACM Press, New York, USA, 2001
- [Bars88] BARSALOU, L.: *The content and organization of autobiographical memories*. In: NEISSER, U. (Hrsg.); WINOGRAD, E. (Hrsg.): *Remembering reconsidered: Ecological and traditional approaches to the study of memory*. Cambridge University Press, 1988, S. 193–243
- [Ber⁺98] BERNERS-LEE, T.; FIELDING, R.; IRVINE, U. C.; MASINTER, L.: *IETF RFC 2396 (Uniform Resource Identifiers)*. <http://www.ietf.org/rfc/rfc2396.txt>, 1998
- [Ber⁺01] BERNERS-LEE, T.; HENDLER, J.; LASSILA, O.: *A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities*. In: *Scientific American*, (2001), May
- [Berg01] BERGMANN, R.: *Experience Management. Habilitationsschrift*, Uni Kaiserslautern, Dissertation, 2001
- [Berg02] BERGMANN, R.: *Experience management - foundations, development methodology, and Internet based applications*. Springer Verlag, Berlin, Heidelberg, 2002
- [Bern98] BERNERS-LEE, T.: *What the Semantic Web can represent*, 1998
- [Bers00] BERST, J.: *Finally, A Silicon Valley Buzzword Worth Remembering: Evernet*. In: *ZDNet AnchorDesk*, (2000), Dez.
- [BeSc04] BECKER, J.; SCHÜTTE, R.: *Handelsinformationssysteme*. Moderne Industrie, 2004
- [BeSe93] BELLOTTI, V.; SELLEN, A.: *Design for Privacy in Ubiquitous Computing Environments*. In: *Proceedings of The Third European Conference on Computer Supported Cooperative Work (ECSCW'93). Milan, Italy*, Kluwer Academic Publishers, 1993

- [BeVo96] BECKER, J.; VOSSEN, G.: *Geschäftsprozeßmodellierung und Workflowmanagement: Eine Einführung*. In: VOSSEN, G. (Hrsg.); BECKER, J. (Hrsg.): *Geschäftsprozeßmodellierung und Workflow-Management – Modelle, Methoden, Werkzeuge*, International Thomsen Publishing, Bonn, 1996, S. 17–26
- [BiCa04] BIEGEL, G.; CAHILL, V.: *A Framework for Developing Mobile, Context-aware Application*. In: *2nd IEEE Conference on Pervasive Computing and Communications, Percom 2004, Orlando, FL, 2004*
- [Bie⁺99] BIEZUNSKI, M.; BRYAN, M.; NEWCOMB, S. R.: *ISO/IEC 13250:2000 Topic Maps: Information Technology - Document Description and Markup Languages*. <http://www.y12.doe.gov/sgml/sc34/document/0129.pdf>, December 1999
- [BiSt99] BILLINGHURST, M.; STARNER, T.: *Wearable Devices: New Ways to Manage Information*. In: *IEEE Computer*, 32 (1999), January 1, S. 57–64
- [BMW95] BMWI: *Die Informationsgesellschaft - Fakten-Analysen - Trends*. In: Report des BMWI, Zeitbild-Verlag, Bonn, (1995), Nov.
- [BMW03] BMWI: *Informationsgesellschaft Deutschland 2006 - Aktionsprogramm der Bundesregierung*, 2003
- [Bode97] BODE, J.: *Der Informationsbegriff in der Betriebswirtschaftslehre*. In: *zfbf, Zeitschrift für betriebswirtschaftliche Forschung*, 49 (1997) 5, S. 449–468
- [Bode99] BODENDORF, F.: *Wirtschaftsinformatik im Dienstleistungsbereich*. Springer-Verlag, Berlin, Heidelberg, 1999
- [Boro02] BORONOWSKY, M.: *Support of Mobile Work by Wearable Computing - WearIT@work*. In: *The 6th Framework Programme of the European Union*, 2002
- [Boyl94] BOYLE, C.: *MetaDoc: An adaptive hypertext reading system*. In: *User Modelling and User-adapted Interaction*, 4 (1994) 1, S. 1–19
- [BrMa02] BRUSILOVSKY, P.; MAYBURY, M. T.: *From Adaptive Hypermedia to the Adaptive Web*. In: *Communications of the ACM*, 45 (2002) 5, S. 30–33
- [Bro⁺97] BROWN, P. J.; BOWEY, J. D.; CHEN, X.: *Context-aware applications: From the laboratory to the marketplace*. In: *IEEE Personal Communications*, 5 (1997) 4, S. 58–64
- [Bru⁺00] BRUMITT, B.; MEYERS, B.; KRUMM, J.; KERN, A.; SHAFER, S.: *EasyLiving: Technologies for Intelligent Environments*. In: THOMAS, P. (Hrsg.); GELLERSEN, H. W. (Hrsg.): *Handheld and Ubiquitous Computing*. Springer Verlag, Berlin, Heidelberg, 2000
- [Brus97] BRUSILOVSKY, P.: *Efficient Techniques for Adaptive Hypermedia*. In: *Intelligent Hypertext: Advanced techniques for the World Wide Web*, 1997

- [Brus98] BRUSILOVSKY, P.: *Adaptive navigation support in educational hypermedia: an evaluation of the ISIS-Tutor*. In: *Journal of Computing and Information Technology*, 6 (1998) 1, S. 27–38
- [Bud⁺91] BUDER, M.; REHFELD, W.; SEEGER, T.: *Grundlagen der praktischen Information und Dokumentation*. K.G. Saur Verlag, München, 1991
- [BuHa00] BUDZIK, J.; HAMMOND, K.: *User interactions with everyday applications as context for just-in-time information access*. In: *Proceedings of Intelligent User Interfaces*, ACM Press, New York, USA, 2000
- [Bühl97] BÜHL, A.: *Die virtuelle Gesellschaft - Ökonomie, Politik und Kultur im Zeichen des Cyberspace*. Westdeutscher Verlag, Opladen, Wiesbaden, 1997
- [CaMi92] CAUDELL, T. P.; MIZELL, D. W.: *Augmented Reality: an application of heads-up display technology to manual manufacturing processes*. In: *Proceedings of the Twenty-Fifth Hawaii International Conference on Systems Science, Kauai, Hawaii*, 1992, S. 659–669
- [Camp00] CAMPBELL, D.: *Inside Echelon*. In: *Telepolis*, <http://www.heise.de/tp/r4/artikel/6/6929/1.html>, (2000)
- [Car⁺94] CARRIERO, N.; GELERNTER, D.; MATTSON, T.; SHERMAN, A.: *The Linda alternative to message-passing systems*. In: *Parallel Computing*, (1994) 20, S. 633–655
- [Che⁺99] CHEVERST, K.; DAVIES, N.; MITCHELL, K.; FRIDAY, A.: *The Role of Connectivity in Supporting Context-Sensitive Applications*. In: *First International Symposium on Handheld and Ubiquitous Computing (HUC99), Karlsruhe, Germany*, Springer Verlag, Berlin, Heidelberg, 1999
- [Che⁺00] CHEVERST, K.; DAVIES, N.; MITCHELL, K.; FRIDAY, A.; EFSTRATIOU, C.: *Experiences of Developing and Deploying a Context-aware Tourist Guide: the GUIDE Project*. In: *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking (MobiCom 2000), Boston*, 2000, S. 20–31
- [Choo95] CHOO, C. W.: *Information Management for the Intelligent Organization: Roles and Implications for the information professions*. In: *Proc of the 1995 digital Libraries Conference, Singapore*, 1995
- [CIMa01] CLARK, J.; MAKOTO, M.: *RELAX NG Specification*. In: *OASIS Committee Specification*, (2001)
- [Coas37] COASE, R. H.: *The Nature of the Firm*. In: *Economica*, 386 (1937) 4
- [Codd70] CODD, E.: *A Relational Model for Large Shared Databanks*. In: *Communications of the ACM*, 13 (1970) 6, S. 377–387

- [Coe⁺99] COEN, M.; PHILLIPS, B.; WARSHAWSKY, N.; WEISMAN, L.; PETERS, S.; FININ, P.: *Meeting the Computational Needs for Intelligent Enironments: the MetaGlue System*. In: *Managing Interactions in Smart Environments, Dublin, Ireland*, Paddy Nixon AND Gerard Lacey AND Simon Dobson, 1999
- [Coh⁺01] COHEN, N. H.; PURAKAYASTHA, A.; TUREK, J.; WONG, L.; YEH, D.: *Challenges in flexible aggregation of pervasive data*. IBM Research, Forschungsbericht, 2001
- [Coy92] COY, P.: *Big Brother Pinned to your Chest*. In: *Business Week*, 38 (1992), August 3279, S. 32–43
- [Curt01] CURTH, M. A.: *Dokumentation*. In: MERTENS, P. (Hrsg.): *Lexikon der Wirtschaftsinformatik*. 4. Auflage. Springer Verlag, Berlin, Heidelberg, New York, 2001, S. 150–152
- [Dav⁺01] DAVIES, N.; CHEVEST, K.; MITCHELL, K.; ENFRAT, A.: *Using and determining location in a Context-sensitive tour guide*. In: *IEEE Computer*, (2001), S. 35–42
- [Dave96] DAVENPORT, T. H.: *Some principles of knowledge management*. <http://knowman.bus.utexas.edu/pubs/kmprin.htm>, 1996
- [DeSo03] DEY, A.; SOHN, T.: *Supporting End User Programming of Context-Aware Applications*. In: *CHI 2003 workshop on End User Development*, 2003, S. 23–26
- [Dey98] DEY, A. K.: *Context-aware Computing: the Cyberdesk project*. In: *Proceedings of the AAAI 1998 Spring Symposium on Intelligent Environments (AAAI Technical Report SS-98-02)*, AAAI Press, (1998), S. 51–54
- [Dey⁺99a] DEY, A.; SALBER, D.; FUTAKAWA, M.; ABOWD, G.: *An Architecture to Support Context-Aware Applications*. In: *12th Ann. ACM Symposium on User Interface Software and Technology (UIST 99)*, Asheville, USA, 1999
- [Dey⁺99b] DEY, A. K.; FUTAKAWA, M.; SALBER, D.; ABOWD, G. D.: *The Conference Assistant: Combining context-awareness with wearable computing*. In: *Proceedings of the 3rd International Symposium on Wearable Computers (ISWC'99)*, San Francisco, 1999, S. 21–28
- [Dey⁺00a] DEY, A.; MANKOFF, J.; ABOWD, G.: *Distributed mediation of imperfectly sensed context in aware environments*. Georgia Institute of Technology, Forschungsbericht GIT-GVU-00-14, 2000
- [Dey00b] DEY, A. K.: *Providing Architectural Support for Building Context-Aware Applications*, Georgia Institute of Technology, Dissertation, Nov. 2000
- [Dey01a] DEY, A.: *Understanding and Using Context*. In: *Personal and Ubiquitous Computing*, (2001), S. 4–13
- [Dey⁺01b] DEY, A. K.; ABOWD, G. D.; SALBER, D.: *A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications*. In: *Human Computer Interaction*, 16 (2001), S. 97–166

- [Dieb96] DIEBOLD, J.: *Die Informationsgesellschaft: Chancen und Risiken für Deutschland*. In: Diebold Management Report, 26 (1996) 2
- [DIN96] DIN: *DIN EN ISO 9241-10 - Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten - Teil 10: Grundsätze der Dialoggestaltung*, 1996
- [DIN04] DIN: *DIN 55350-11: Begriffe zu Qualitätsmanagement und Statistik - Teil 11: Begriffe des Qualitätsmanagements*, 2004
- [Dobs03] DOBSON, S.: *Applications considered harmful for ambient systems*. In: *Proceedings of the International Symposium on Information and Communication Technologies*, Trinity College, Dublin, Ireland, 2003
- [DoHu03] DORNBUSCH, P.; HUBER, M.: *Generierung von Ortsinformationen durch User-Communities*. In: *Proceedings der WI 2003*, 2003
- [DoZü02] DORNBUSCH, P.; ZÜNDT, M.: *Realisierung von Positionsortungen in WLAN*. In: *ITG-Fachtagung "Technologie und Anwendungen für die mobile Informationsgesellschaft"*, Dresden, 2002
- [Dres99] DRESBACH, S.: *Epistemologische Überlegungen zu Modellen in der Wirtschaftsinformatik*. In: BECKER, J. (Hrsg.); KÖNIG, W. (Hrsg.); SCHÜTTE, R. (Hrsg.); WENDT, O. (Hrsg.); ZELEWSKI, S. (Hrsg.): *Wirtschaftsinformatik und Wissenschaftstheorie. Bestandsaufnahme und Perspektiven*. Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, Wiesbaden, 1999, S. 71–94
- [Ebl⁺01] EBLING, M.; HUNT, G.; LEI, H.: *Issues for context services for pervasive computing*. In: *Middleware 2001. Workshop on Middleware for Mobile Computing*, 2001
- [ECMA99] ECMA: *ECMAScript Language Specification*, 1999
- [Efs⁺01] EFSTRATIOU, C.; CHEVERST, K.; DAVIES, N.; FRIDAY, A.: *An architecture for the effective support of adaptive context-aware applications*. In: *Mobile Data Management (MDM)*, Hong Kong, China, 2001, S. 15–26
- [EiKI01] EISENHAUER, M.; KLEMKE, R.: *Contextualisation in Nomadic Computing*. In: *ERCIM News*, Special Theme: Ambient Intelligence, (2001), Oktober 47
- [Eld⁺94] ELDRIDGE, M.; BARNARD, P.; BEKERIAN, D.: *Autobiographical memory and daily schemas at work*. In: *Memory*, 1994
- [FaCI04] FAHY, P.; CLARKE, S.: *CASS - Middleware for Mobile Context-Aware Applications*. In: *Proceedings of the MobiSys 2004 Workshop on Context Awareness*, 2004
- [FaWa04] FALLSIDE, D. C.; WALMSLEY, P.: *XML Schema Part 0: Primer Second Edition*. In: *W3C Recommendation*, (2004)

- [Fay⁺96] FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P.: *The KDD Process for Extracting Useful Knowledge from Volumes of Data*. In: Communications of the ACM, 39 (1996) 11, S. 27–34
- [FeDa95] FELDMAN, R.; DAGAN, I.: *Knowledge Discovery in Textual Databases*. In: *First International Conference on Knowledge Discovery and Data Mining*, 1995, S. 112–117
- [Fehr04] FEHRE, T.: *Konzeption einer Lösung zur Darstellung und Speicherung von Kontext-Metadaten unter Berücksichtigung semantischer Zusammenhänge*, TU-Dresden, Lehrstuhl Rechnernetze, Fakultät Informatik, Diplomarbeit, 2004
- [Fei⁺93] FEINER, S.; MACINTYRE, B.; SELIGMANN, D.: *Knowledge-based Augmented Reality*. In: Communications of the ACM, 36 (1993), July 7, S. 65–75
- [Fein00] FEINMAN, J. M.: *Law 101*. Oxford University Press, New York, Oxford, 2000
- [Fenn99] FENN, J.: *Skill Mining: An Emerging KM Technology*. In: *Gartner Group Report*, 1999
- [FeSi84] FERSTL, O. K.; SINZ, E. J.: *Software-Konzepte der Wirtschaftsinformatik*. Verlag DeGruyter, Berlin, 1984
- [FeSi98] FERSTL, O. K.; SINZ, E. J.: *Grundlagen der Wirtschaftsinformatik, Band 1*. Oldenbourg Wissenschaftsverlag GmbH, München, Wien, 1998
- [FeZa00] FEMPEL, K.; ZANDER, E.: *Leistungsorientierte Vergütung*. Bachem Wirtschaft, 2000
- [Fia⁺03] FIALA, Z.; HINZ, M.; MEISSNER, K.; WEHNER, F.: *A Component-based Approach for Adaptive, Dynamic Web Documents*. In: Journal of Web Engineering, 2 (2003) 1 & 2
- [Fisc03] FISCHMEISTER, S.: *Mobile Software Agents for Location-based Systems*. In: *Agent Technologies, Infrastructures, Tools and Applications for E-Services: NODe 2002 Agent-Related Workshops, Erfurt*, 2003, S. 226–239
- [Fit⁺02] FITZPATRICK, A.; BIEGEL, G.; CLARKE, S.; CAHILL, V.: *Towards a Sentient Object Model*. In: *Int. Workshop on Engineering Context-Aware OO Software*, 2002
- [Fitz93] FITZMAURICE, G.: *Situated Information Spaces and Spatially Aware Palmtop Computers*. In: Communications of the ACM, (1993)
- [FIDi03] FLEISCH, E.; DIERKES, M.: *Ubiquitous Computing aus betriebswirtschaftlicher Sicht*. In: Wirtschaftsinformatik, 6 (2003) 45, S. 611–620
- [Flec80] FLECK, L.: *Entstehung und Entwicklung einer wissenschaftlichen Tatsache*. Suhrkamp, Frankfurt/M., 1980
- [Flec83] FLECK, L.: *Erfahrung und Tatsache*. stw 404, Suhrkamp, Frankfurt/M., 1983
- [FoFo04] FORMAN, I. R.; FORMAN, N.: *Java Reflection in Action*. Manning Publications, 2004

- [Fors99] FORST, A.: *Information und Wissen: Die neuen betrieblichen Ressourcen*. In: doculine News, (1999), Februar
- [Fox⁺98] FOX, A.; GRIBBLE, S. D.; CHAWATHE, Y.; BREWER, E. A.: *Adapting to Network and Client Variation Using Active Proxies: Lessons and Perspectives*. In: IEEE Personal Communications, (1998), S. 10–19
- [Fra⁺01] FRANCE, E.; TRACEY, K.; NEUREITER, G.; PASCOTTO, R.; SERRANO, M.; PARKKILA, S.; PIRINEN, J.: *Where are the other mobile buddies around town?* In: *IST Mobile Summit 2001*, 2001
- [Fuc⁺76] FUCHS-KITTOWSKI, K.; KAISER, H.; TSCHIRSCHWITZ, R.; WENZLAFF, B.: *Informatik und Automatisierung*. Berlin, 1976
- [Fuc⁺99] FUCHS-KITTOWSKI, K.; HEINRICH, L.; ROLF, A.: *Information entsteht in Organisationen in kreativen Unternehmen. Wissenschaftstheoretische und methodologische Konsequenzen für die Wirtschaftsinformatik*. In: BECKER, J. (Hrsg.); KÖNIG, W. (Hrsg.); SCHÜTTE, R. (Hrsg.); WENDT, O. (Hrsg.); ZELEWSKI, S. (Hrsg.): *Wirtschaftsinformatik und Wissenschaftstheorie. Bestandsaufnahme und Perspektiven*. Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, Wiesbaden, 1999, S. 329–362
- [Fuch97] FUCHS-KITTOWSKI, K.: *Der Mensch muß in hochkomplexen Informationstechnologischen Systemen höchste Autorität sein und bleiben*. In: Hochschulforum, Brandenburger Tagungsberichte, (1997)
- [FuRo98] FUCHS-KITTOWSKI, K.; ROSENTHAL, H.-A.: *Selbstorganisation, Information und Evolution - zur Kreativität der belebten Natur*, 1998
- [Gaj⁺02] GAJOS, K.; FOX, H.; SHROBE, H.: *End User Empowerment in Human Centered Pervasive Computing*. In: *Proceedings of Pervasive 2002*, 2002
- [Gell99] GELLERSEN, H.-W.: *Ubiquitäre Informationstechnologien*. <http://www.vs.inf.ethz.ch/vs/publ/papers/UbiqGell.pdf>, 1999
- [Geri00] GERICK, T.: *Sinnsuche. Retrieval: Methoden, Trends, Produkte*. In: iX, Heise Zeitschriften Verlag, (2000)
- [GeTe01] GELDOF, S.; TERKEN, J.: *Talking Wearables Exploit Context*. In: *Personal and Ubiquitous Computing*, 5 (2001) 1, S. 62–65
- [GoHo00] GOESMANN, T.; HOFFMANN, M.: *Unterstützung wissensintensiver Geschäftsprozesse durch Workflow-Management-Systeme*. In: *DCSCW 2000*, Springer Verlag, Berlin, Heidelberg, 2000
- [Gos⁺03] GOSLAR, K.; BUCHHOLZ, S.; SCHILL, A.; VOGLER, H.: *A Multidimensional Approach to Context-Awareness*. In: CALLAOS, N. (Hrsg.): *Proc. of the 7th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2003)*, 2003, S. 229–234

- [Gosl01] GOSLAR, K.: *Gewinnung, Darstellung und Verwaltung von Kontextinformationen – Personalisierung*, TU-Dresden, Diplomarbeit, 2001
- [Gosl04] GOSLAR, K.: *Modelling Contextual Information using Active Data Structures*. In: TÜRKER, C. (Hrsg.): *Proc. of Intl. Workshop on Pervasive Information Management*, 2004, S. 12–18
- [GrHo03] GRUDIN, J.; HORVITZ, E.: *Presenting choices in context: approaches to information sharing*. In: *Workshop on Ubicomp communities: Privacy as Boundary Negotiation*, 2003
- [Groc75] GROCHLA, E.: *Betriebliche Planung und Informationssysteme – Entwicklung und aktuelle Aspekte*. Rowohlt, Reinbeck, 1975
- [Grou97] GROUP, D.: *Delphi On Knowledge Management – Research & Perspectives On Today's Knowledge Landscape*. Delphi Group, 1997
- [Grou00] GROUP, G.: *Mobilising the Real-Time Enterprise*, 2000
- [Grou03] GROUP, M.: *Metro Group – Future Store Initiative*. <http://www.future-store.org>, 2003
- [GrSa01] GRAY, P.; SALBER, D.: *Modelling and using sensed context in the design of interactive applications*. In: *8th IFIP Conference on Engineering for Human-Computer Interaction, Toronto*, 2001
- [Grub93a] GRUBER, T.: *Toward principles for the design of ontologies used for knowledge sharing*. In: GUARINO, N. (Hrsg.); POLI, R. (Hrsg.): *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers, 1993
- [Grub93b] GRUBER, T. R.: *A Translation Approach to Portable Ontology Specifications*. In: *Knowledge Acquisition*, 5 (1993) 2, S. 199 – 220
- [Guar98] GUARINO, N.: *Formal ontology in information systems*. In: *Proc of FOIS'98, Trento, Italy*, 1998
- [Gute51] GUTENBERG, E.: *Grundlagen der Betriebswirtschaftslehre*. Springer-Verlag, Berlin, Heidelberg, 1951
- [Haas96] HAAS, C.: *Prof. Brunnstein und seine gesammelten Pannen: Spaß (Fun) am Absturz von Computern und Netzen?* In: *Die Datenschleuder - das wissenschaftliche Fachblatt für Datenreisende - Das offizielle Organ des Chaos Computer Clubs*, (1996) 54
- [Häc⁺00] HÄCKELMANN, H.; PETZOLD, H. J.; STRAHRINGER, S.: *Kommunikationssysteme – Technik und Anwendungen*. Springer Verlag, Berlin, Heidelberg, 2000
- [Haef80] HAEFNER, K.: *Der 'große Bruder'. Chancen und Gefahren für eine informierte Gesellschaft*. Econ, Düsseldorf, 1980

- [Han⁺02] HANSEN, M.; KÖHNTOPP, K.; PFITZMANN, A.: *The Open Source approach - opportunities and limitations with respect to security and privacy*. In: *Computers & Security*, 21 (2002) 5, S. 461–471, http://dud.inf.tu-dresden.de/literatur/HaKP_02OpenSource_0214.pdf
- [Hanf91] HANF, V.: *Integrierte Datenmodellierung bei BMW - ein Erfahrungsbericht*. In: *Wirtschaftsinformatik*, 33 (1991) 4, S. 300–307
- [Har⁺97] HARTLEY, R.; KELSEY, R.; WEBSTER, R.: *Consolidating Multi-Source and Multi-Media Knowledge*. In: *AAAI Spring Symposium on Artificial Intelligence in Knowledge Management*, 1997
- [Har⁺01] HARTER, A.; HOPPER, A.; STEGGLES, P.; WARD, A.; WEBSTER, P.: *The Anatomy of a Context-Aware Application*. In: *Wireless Networks*, 1 (2001) 1, S. 1–16
- [Har⁺03] HARVEL, L.; LUI, L.; ABOWD, G.; LIM, Y.-X.; SCHEIBE, C.; CHATHAM, C.: *Leveraging History: Building Context Cubes for Context-Aware Applications*. Georgia Institute of Technology, Forschungsbericht, 2003
- [Harp95] HARPER, R. H.: *Why people do and don't wear active badges: a case study*. In: *Computer Supported Cooperative Work*, 4 (1995) 4, S. 297 – 318
- [Hart98] HARTKOPF, V.: *Cooperative Workplaces and Buildings*. In: *Cooperative Buildings - Integrating Information, Organization and Architecture*, 1998
- [HaSc94] HARS, A.; SCHEER, A. W.: *Paradigmenwechsel im Informationsmanagement: Vom DV-Management zum Management des Produktionsfaktors Information*. In: *IM, Fachzeitschrift für Information, Management und Consulting*, 9 (1994) 2, S. 6–11
- [HaSc04] HAMMEL, H.; SCHWOTZER, T.: *Gewusst wo - Ortsabhängige Topic Maps*. In: *iX - Magazin für professionelle Informationstechnik*, (2004) 02
- [Heer01] HEEREN, F.: *Der direkte Weg zum Wissen in den Köpfen*. In: *Wissensmanagement*, (2001) 04
- [Hefl01] HEFLIN, J. D.: *Towards the Semantic Web: Knowledge Representation in a Dynamic, Distributed Environment*, University of Maryland, College Park, Dissertation, 2001
- [Hei⁺96] VAN HEIJST, G.; VAN DER SPEK, R.; KRUIZINGA, E.: *Organising Corporate Memories*. In: *Proc of the 10th Knowledge Acquisition for Knowledge-Based Systems Workshop*, 1996
- [Hein93] HEINRICH, L. J.: *Wirtschaftsinformatik: Einführung und Grundlegung*. Oldenbourg, München, 1993
- [Hein99] HEINRICH, L. J.: *Informationsmanagement - Planung, Überwachung und Steuerung der Informationsinfrastruktur*. Oldenbourg, München, Wien, 1999

- [Hen⁺02] HENRICKSEN, K.; INDULSKA, J.; RAKOTONIRAINY, A.: *Modeling Context Information in Pervasive Computing Systems*. In: MATTERN, F. (Hrsg.); NAGHSHINEH, M. (Hrsg.): *Pervasive 2002. LNCS 2414*, Springer Verlag, Berlin, Heidelberg, 2002, S. 167–180
- [Henz92] HENZLER, R. G.: *Information und Dokumentation*. Springer Verlag, Berlin, Heidelberg, New York, 1992
- [Her⁺59] HERZBERG, F.; MAUSNER, B.; SNYDERMAN, B.: *The Motivation to Work*. Wiley, New York, 1959
- [Hes⁺02] HESS, C. K.; ROMAN, M.; COMPELL, R. H.: *Building applications for ubiquitous computing environments*. In: *Proc. of the International Conference on Pervasive Computing (Pervasive 2002)*, Zurich, Switzerland, 2002
- [HeSi01] HEEREN, F.; SIHN, W.: *XPRTFINDER – Nachrichtenanalyse zur themenfeldbezogenen Expertensuche*. In: GRONAU, N. (Hrsg.): *Wissensmanagement: Systeme - Anwendungen - Technologien*, Shaker Verlag, Aachen, 2001, S. 81–92
- [HiSc92] HINDUS, D.; SCHMANDT, C.: *Ubiquitous Audio: Capturing Spontaneous Collaboration*. In: *Proceedings of the 1992 ACM conference on Computer-supported cooperative work*, 1992
- [Hod⁺97] HODES, T. D.; KATZ, R. H.; SERVAN-SCHREIBER, E.; ROWE, L.: *Composable Ad-hoc Mobile Services for Universal Interaction*. In: *Proceedings of the Third ACM/IEEE International Conference on Mobile Computing (MobiCom 97)*, 1997, S. 1–17
- [HoLa01] HONG, J. I.; LANDAY, J.: *Integrating Context Services Through Automatic Path Creation*. In: *CHI2001 Workshop for Context-Awareness*, 2001
- [Holl99] HOLL, A.: *Empirische Wirtschaftsinformatik und Erkenntnistheorie*. In: BECKER, J. (Hrsg.); KÖNIG, W. (Hrsg.); SCHÜTTE, R. (Hrsg.); WENDT, O. (Hrsg.); ZELEWSKI, S. (Hrsg.): *Wirtschaftsinformatik und Wissenschaftstheorie. Bestandsaufnahme und Perspektiven*. Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, Wiesbaden, 1999, S. 163–209
- [Hon⁺03] HONG, J. I.; BECKMANN, C.; HEER, J.; NEWBERGER, A.: *Confab Tutorial*, 2003
- [Hong01] HONG, J.: *Context Fabric: Infrastructure Support for Context-Aware Systems*. University of California at Berkeley, Qualifying Exam Proposal, 2001
- [Hong02] HONG, J. I.: *The Context Fabric: An Infrastructure for Context-Aware Computing*. In: *CHI 2002*, 2002
- [Hong04] HONG, J.: *Designing Privacy Risk Models*. In: *DIS 2004*, 2004
- [Horr02] HORROCKS, I.: *DAML+OIL: a description logic for the semantic web*. In: *IEEE Bulletin Tech Comm Data Engine*, 25 (2002) 1, S. 4–9

- [Hous71] HOUSE, R.: *A path-goal theory of leader effectiveness*. In: *Administrative Science Quarterly*, (1971) 16, S. 321–339
- [Houy96] HOUY, C.: *Datenmanagement für Workflowprozesse. Vorbereitung und Unterstützung beim Austausch operativer Daten im industriellen Umfeld*. Gabler, Wiesbaden, 1996
- [Hul⁺97] HULL, R.; NEAVES, P.; BEDFORD-ROBERTS, J.: *Towards Situated Computing*. In: *Proc. of The First Int. Symposium on Wearable Computers, Cambridge, Massachusetts*, IEEE Computer Society Press, 1997
- [Huss86] HUSSERL, E.; HELD, K. (Hrsg.): *Edmund Husserl. Phänomenologie der Lebenswelt. Ausgewählte Texte II*. Reclam, Ditzingen Reclam, Ditzingen, 1986
- [Hut⁺01] HUTH, C.; SMOLNIK, S.; NASTANSKY, L.: *Applying Topic Maps to Ad Hoc Workflows for Semantic Associative Navigation in Process Networks*. In: *Seventh International Workshop on Groupware (CRIWG'01)*, 2001, S. 44–49
- [IBM03] IBM: *IBM Autonomous Computing Manifesto*, 2003
- [Ihde01] IHDE, G. B.: *Transport, Verkehr, Logistik*. Vahlen-Verlag, München, 2001
- [II92] II, B. J. P.: *Mass Customization: The New Frontier in Business Competition*. Harvard Business School Press, 1992
- [Inmo95] INMON, B.: *From Transactions to the Operational Data Store*. In: *InfoDB*, 9 (1995) 6, S. 11–13
- [Inmo99] INMON, W. H.: *Building the Operational Data Store*. John Wiley & Sons, 1999
- [JaBu96] JABLONSKI, S.; BUSSLER, C.: *Workflow Management. Modeling Concepts, Architecture and Implementation*. Informational Thomson Computer Press, 1996
- [Jak⁺03] JAKSCH, S.; PFENNIGSCHMIDT, S.; SANDKUHL, K.; THIEL, C.: *Information logistic applications for information-on-demand scenarios: concepts and experiences from WIND project*. In: *Proc. of the 29th Conference on EUROMICRO* IEEE Computer Society, 2003
- [Jan⁺01] JANG, S.; KIM, J.; RARNAKRISHNA, R. S.: *Framework for Building Mobile Context-Aware Applications*. In: ET AL., W. K. (Hrsg.): *Human Society @ Internet*, 2001, S. 139–150
- [Jons02] JONSSON, M.: *Context Shadow: A Person-Centric Infrastructure for Context Aware Computing*. In: *Workshop on Artificial Intelligence in Mobile Systems*, 2002
- [Kadn04a] KADNER, K.: *Erweiterung eines verteilten Kontextdienstes um Konzepte zur Unterstützung von ad hoc Szenarien*, Technische Universität Dresden, Diplomarbeit, 2004
- [Kadn04b] KADNER, K.: *Konzeption eines verteilten Dienstes zur Unterstützung von Context-Awareness*, TU-Dresden, Diplomarbeit, 2004

- [Kah⁺99] KAHN, J.; KATZ, R.; PISTER, K.: *Mobile Networking for Smart Dust*. In: *Proc of MobiCom 1999: The fifth Annual Int. Conference on Mobile Computing and Networking*, ACM Press, New York, USA, 1999, S. 271–278
- [Kap193] KAPLAN, C.: *Adaptive hypertext navigation based on user goals and context*. In: *User Modelling and User-Adapted Interaction*, 3 (1993) 3, S. 193–220
- [KeCh03] KEPHART, J. O.; CHESS, D. M.: *The Vision of Autonomic Computing*. In: *Computer Magazine*, http://researchweb.watson.ibm.com/autonomic/research/papers/AC_Vision_Computer_Jan_2003.pdf, (2003), January
- [Kemk88] KEMKE, C.: *Der neuere Konnektionismus. Ein Überblick*. In: *Informatik Spektrum*, 11 (1988) 3, S. 143–162
- [Kes⁺96] KESDOGAN, D.; FEDERRATH, H.; JERICHOW, A.; PFITZMANN, A.: *Location Management Strategies increasing Privacy in Mobile Communication Systems*. In: *IFIP SEC '96, Proceedings of the IFIP TC11*, Chapman & Hall, London, 1996, S. 39–48
- [Klem99] KLEMKE, R.: *The Notion of Context in Organizational Memories*. In: *Proceedings of Modeling and Using Context*, Springer Verlag, Berlin, Heidelberg, 1999, S. 483–486
- [Klem00] KLEMKE, R.: *Context Framework - an Open Approach to Enhance Organisational Memory Systems with Context Modelling Techniques*. In: *PAKM2000: Third International Conference on Practical Aspects of Knowledge Management*, 2000
- [Kolo91] KOLODNER, J. L.: *Improving human decision making through case-based decision aiding*. In: *AI magazine*, 12 (1991) 2
- [Kön⁺95] KÖNIG, W.; HEINZL, A.; POBLOTZKI, A.: *Die zentralen Forschungsgegenstände der Wirtschaftsinformatik in den nächsten zehn Jahren*. In: *Wirtschaftsinformatik*, 6 (1995) 37, S. 558–569
- [Kor⁺98] KORTUEM, G.; SEGALL, Z.; BAUER, M.: *Context-Aware, Adaptive Wearable Computers as Remote Interfaces to 'Intelligent' Environments*. In: *2nd International Symposium on Wearable Computers*, 1998, S. 58–67
- [Kosk91] KOSKY, A.: *Modeling and Merging Database Schemas*. University of Pennsylvania, Forschungsbericht Technical Report MS-CIS-91-65, 1991
- [Krcm92] KRCCMAR, H.: *Informationslogistik der Unternehmung*. In: STROETMANN, K. A. (Hrsg.): *Informationslogistik*. DGD, Frankfurt a. M., 1992
- [Krcm97] KRCCMAR, H.: *Informationsmanagement*. Springer Verlag, Berlin, Heidelberg, 1997
- [Kres01] KRESSLER, H.: *Leistungsbeurteilung und Anreizsysteme*. Ueberreuter Wirtschaft, 2001

- [KrSi99] KRÜGER, R.; SIMON, A.: *Information durch Konstruktion - Systemtheoretische Betrachtung des Phänomens Informationsverarbeitung und wissenschaftstheoretische Schlußfolgerung für die Wirtschaftsinformatik*. In: BECKER, J. (Hrsg.); KÖNIG, W. (Hrsg.); SCHÜTTE, R. (Hrsg.); WENDT, O. (Hrsg.); ZELEWSKI, S. (Hrsg.): *Wirtschaftsinformatik und Wissenschaftstheorie. Bestandsaufnahme und Perspektiven*. Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, Wiesbaden, 1999, S. 363–376
- [KuAb97] KUEHN, O.; ABECKER, A.: *Corporate Memories for Knowledge Menegement in Industrial Practive: Prospects and Challenges*. In: *Journal of Universal Computer Science*, 8 (1997) 3, S. 929–954
- [Kuhn88a] KUHN, T. S.: *Die Entstehung des Neuen*. 3. Auflage, Suhrkamp, Frankfurt/M., 1988
- [Kuhn88b] KUHN, T. S.: *Die Struktur wissenschaftlicher Revolutionen*. 9. Auflage, Suhrkamp, Frankfurt/M., 1988
- [Kutz03] KUTZENBERGER, E.: *Leistungs- und Strukturhebung – Produktion und Dienstleistungen*. Statistik Austria, Wien, 2003
- [Lam⁺94] LAMMING, M. G.; BROWN, P. J. et al.: *The Design of a Human Memory Prosthesis*. In: *Computer Journal*, 37 (1994), January 3, S. 153–163
- [Lamm91] LAMMING, M. G.: *NoTime – A Tool for Notetakers*. EuroPARC, Forschungsbericht, 1991
- [LaNe92] LAMMING, M. G.; NEWMAN, W. M.: *Activity-based Information Retrieval: Technology in Support of Personal Memory*. In: *Proceedings of the IFIP 12th World Computer Congress on Personal Computers and Intelligent Systems - Information Processing '92*, North-Holland, 1992, S. 68 – 81
- [Lang01] LANGHEINRICH, M.: *Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems*. In: *Proceedings of Ubicomp*, 2001, S. 273–291
- [Lehn99] LEHNER, F.: *Theoriebildung in der Wirtschaftsinformatik*. In: ET AL., J. B. (Hrsg.): *Wirtschaftsinformatik und Wissenschaftstheorie. Bestandsaufnahme und Perspektiven*. Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, Wiesbaden, 1999, S. 5–24
- [Lei⁺00] LEIBE, B.; STARNER, T.; RIBARSKY, W.; WARTELL, Z.; KRUM, D.; WEEKS, J.; SINGLETARY, B.; HODGES, L.: *Towards Spontaneous Interaction with the Perceptive Workbench*. In: *IEEE Computer Graphics and Applications*, 20 (2000) 6
- [Lei⁺02] LEI, H.; SOW, D. M.; II, J. S. D.: *The Design and Application of a Context Service*. In: *ACM SIGMOBILE Mobile Computing and Communications Review*, 6 (2002) 4, S. 45–55
- [LeKr02] LEIMEISTER, J. M.; KRCMAR, H.: *Ubiquitous Computing*. In: *wisu - Das Wirtschaftsstudium*, 31 (2002) 10, S. 1284–1294

- [Lena95] LENANT, D. B.: *Cyc: A Large-Scale Investment in Knowledge Infrastructure*. In: *Communications of the ACM*, 38 (1995) 11
- [Leo⁺96] LEONHARD, U.; MAGEE, J.; DIAS, P.: *Location service in mobile computing environments*. In: *Computer & Graphics*, 20 (1996) 5, S. 627–632
- [Leo⁺99] LEONHARDI, A.; KUBACH, U.; ROTHERMEL, K.; FRITZ, A.: *Virtual Information Towers - A Metaphor for Intuitive, Location-Aware Information Access in a Mobile Environment*. In: *Proceedings of the Third International Symposium on Wearable Computers, San Francisco, 1999*, S. 15–20
- [Li⁺00] LI, J.; JANOTTI, J.; COUTO, D. D.; KARGER, D.; MORRIS, R.: *A Scalable Location Service for Geographic Ad Hoc Routing*. In: *Mobicom 2000, Boston, 2000*
- [Lie⁺99] LIECHTI, O.; SIEFER, N.; ICHIKAWA, T.: *Personal Technologies*. In: *A Nonobtrusive User Interface for Increasing Social Awareness on the World Wide Web*, 3 (1999) 1 & 2
- [Lösc02] LÖSCHAU, F.: *Realisierung kontextsensitiver Anwendungen auf Basis eines Frameworks zur Kontextverarbeitung und -verwaltung*, TU-Dresden, Diplomarbeit, 2002
- [Lyo⁺03] LYONS, K.; STARNER, T.; PLAISTED, D.; FUSIA, J.; LYONS, A.; DREW, A.; LOONEY, E. W.: *Twiddler Typing: One-Handed Chording Text Entry for Mobile Phones*. College of Computing and GVU Center, Georgia Institute of Technology, Forschungsbericht, 2003
- [MaJo03] MANTORO, T.; JOHNSON, C.: *Location History in a Low-cost Context Awareness Environment*. In: *Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003 - Volume 21, 2003*, S. 153 – 158
- [Man⁺99] MANIATIS, P.; ROUSSOPOULOS, M.; SWIERK, E.; LAI, K.; APPENZELLER, G.; ZHAO, X.; BAKER, M.: *The Mobile People Architecture*. In: *ACM Mobile Computing and Communications Review*, ACM Press, New York, USA, 1999
- [Manb00] MANBER, U.: *Experience with personalization on Yahoo!* In: *Communications of the ACM*, 43 (2000) 8
- [Mat⁺99] MATTOX, D.; MAYBURY, M.; MOREY, D.: *Enterprise Expert and Knowledge Discovery*. In: *Proceedings of the 8th International Conference on Human-Computer Interaction, 1999*, S. 303–307
- [Matt04] MATTERN, F.: *Ubiquitous Computing: Szenarien einer informatisierten Welt*. In: ZERDICK, A. (Hrsg.); PICOT, A. (Hrsg.); SCHRAPE, K. (Hrsg.); BURGELMAN, J.-C. (Hrsg.); SILVERSTONE, R. (Hrsg.); FELDMANN, V. (Hrsg.); HEGER, D. K. (Hrsg.); WOLFF, C. (Hrsg.): *E-Merging Media - Kommunikation und Medienwirtschaft der Zukunft*, Springer-Verlag, Berlin, Heidelberg, 2004, S. 155–174

- [Maus01] MAUS, H.: *Workflow Context as a Means for Intelligent Information Support*. In: ET AL., V. A. (Hrsg.): *3rd Int. Conference on Modeling and Using Context (CONTEXT 2001)*, Springer Verlag, Berlin, Heidelberg, 2001, S. 261–274
- [McC⁺55] MCCARTHY, J.; MINSKY, M. L.; ROCHESTER, N.; SHANNON, C.: *A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence*. In: *Proceedings of the Dartmouth Conference*, 1955
- [McCr00] MCCRORY, A.: *Ubiquitous? Pervasive? Sorry, they don't compute*. In: *Computerworld*, (2000)
- [McGr60] MCGREGOR, D.: *The Human side of the Enterprise*. McGraw-Hill Book Company, New York, 1960
- [McHa68] MCCARTHY, J.; HAYES, P. J.: *Some philosophical problems from the standpoint of artificial intelligence*. Stanford University, Department of Computer Science, Stanford, California, Forschungsbericht Report Memo AI-73, 1968
- [Mei⁺91] MEIER, A.; GRAF, H.; SCHWIMM, K.: *Ein erster Schritt zu einem globalen Datenmodell*. In: *Information Management*, 6 (1991) 2, S. 42–48
- [Mei⁺04] MEISSEN, U.; PFENNIGSCHMIDT, S.; VOISARD, A.; WAHNFRIED, T.: *Context- and Situation-Awareness in Information Logistics*. In: TÜRKER, C. (Hrsg.): *Proc. of Intl. Workshop on Pervasive Information Management*, 2004, S. 19–30
- [Men⁺98] MENA, E.; KASHYAP, V.; ILLARAMENDI, A.; SHETH, A.: *Domain-Specific Ontologies for Semantic Information Brokering on the Global Information Infrastructure*. In: *Proc. of the First International Conference on Formal Ontologies in Information Systems*, 1998
- [Mer⁺93] MERTENS, P.; BORKOWSKI, V.; GEIS, W.: *Betriebliche Expertensystem-Anwendungen*. 3. Auflage, Berlin, 1993
- [Mer⁺01] MERTENS, P.; BODENDORF, F.; KÖNIG, W.; PICOT, A.; SCHUMANN, M.: *Grundzüge der Wirtschaftsinformatik*. Springer Verlag, Berlin, Heidelberg, 2001
- [Merk95] MERKEL, H.: *Logistik-Managementsysteme*. Oldenbourg Wissenschaftsverlag GmbH, München, Wien, 1995
- [Mert90] MERTENS, P.: *Einsatzpotentiale und Anwendungsklassen für Expertensysteme*. In: KURBEL, K. (Hrsg.); STRUNZ, H. (Hrsg.): *Handbuch Wirtschaftsinformatik*. Stuttgart, 1990, S. 523–540
- [Mert95] MERTENS, P.: *Wirtschaftsinformatik - Von den Moden zum Trend*. In: KÖNIG, W. (Hrsg.): *Wirtschaftsinformatik '95 - Wettbewerbsfähigkeit, Innovation, Wirtschaftlichkeit*, Physica-Verlag, Heidelberg, 1995, S. 25–64

- [Mert03] MERTENS, P.: *Die Wirtschaftsinformatik auf dem Weg zur Unternehmensspitze – alte und neue Herausforderungen und Lösungsansätze*. In: *WI 2003 – Medien, Märkte, Mobilität*. Physica-Verlag, Heidelberg, 2003, S. 49–74
- [Merz02] MERZ, M.: *E-Commerce und E-Business, Marktmodelle, Anwendungen und Technologien*. dpunkt, Heidelberg, 2002
- [Mess99] MESSER, B.: *Zur Interpretation formaler Geschäftsprozess- und Workflow-Modelle*. In: BECKER, J. (Hrsg.); KÖNIG, W. (Hrsg.); SCHÜTTE, R. (Hrsg.); WENDT, O. (Hrsg.); ZELEWSKI, S. (Hrsg.): *Wirtschaftsinformatik und Wissenschaftstheorie. Bestandsaufnahme und Perspektiven*. Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, Wiesbaden, 1999, S. 95–124
- [MeZa96] MEYER, M. H.; ZACK, M. H.: *The Design and Development of Information Products*. In: *Sloan Management Review*, 37 (1996) 3, S. 43–59
- [Mier03] MIERDORF, Z.: *Innovationsführerschaft im Handel – das Beispiel der METRO Group*. In: *WI 2003 - Medien, Märkte, Mobilität*, 2003
- [Mino01] MINOR, M.: *Experience Management - Case Based Reasoning for Knowledge Management*. In: CZAJA, L. (Hrsg.): *Proceedings of CSuP-2001*, 2001, S. 150–159
- [Morr38] MORRIS, C. H.: *Foundation of the theory of signs*. In: *International Encyclopedia of Unified Science* Band 2. University of Chicago Press, 1938
- [Morr91] MORRIS, M.: *Configuration Management: Getting the Facts Right*. In: *Inform*, 5 (1991), May
- [Müll89] MÜLLER-ETTRICH, G.: *Effektives Datendesign*. Rudolf Müller, Köln, 1989
- [Müll03] MÜLLER, G.: *Geduldige Technologie für ungeduldige Patienten*. In: MATTERN, F. (Hrsg.): *Total vernetzt - Szenarien einer informatisierten Welt*, Springer-Verlag, Berlin, Heidelberg, 2003
- [Nard93] NARDI, B. A.: *A Small Matter of Programming: Perspectives on End User Computing*. MIT Press, Cambridge, USA, 1993
- [Nast89] NASTANSKY, L.: *Flexibles Informationsmanagement für Organisatoren mit Werkzeugumgebungen für Persönliches Informationsmanagement (PIM)*. In: *GI Jahrestagung (2)*, 1989, S. 232–245
- [Nast90] NASTANSKY, L.: *Lotus-Agenda-Anwendungen – Personal Information Management Anwendungen auf der Grundlage von Agenda Templates*. In: *Kurzbulletins des Institut für Wirtschaftsinformatik und Informatikbereich, Hochschule St. Gallen*, 1990
- [Nast92] NASTANSKY, L.: *Workgroup Computing*. In: *Betriebswirtschaft aktuell Band 12*, S+W Verlag, 1992

- [Nast95] NASTANSKY, L.: *Büroinformationssysteme*. In: FISCHER, J. (Hrsg.); HEROLD, W. (Hrsg.); W. DANGELMAIER, W. (Hrsg.); NASTANSKY, L. (Hrsg.); WOLFF, R. (Hrsg.): *Bausteine der Wirtschaftsinformatik. Grundlagen, Anwendungen, PC-Praxis*. Steuer- und Wirtschaftsverlag, Hamburg, 1995, S. 267–369
- [New⁺02] NEWMAN, M. W.; SEDIVY, J.; EDWARDS, W.; SMITH, T.; MARCELO, K.; NEUWIRTH, C. M.; HONG, J. I.; IZADI, S.: *Designing for Serendipity: Supporting End-User Configuration of Ubiquitous Computing Environments*. In: *Proceedings of Designing Interactive Systems (DIS2002)*, 2002
- [Norm98] NORMAN, D. A.: *The Invisible Computer*. MIT Press, Cambridge, USA, 1998
- [Nort98] NORTH, K.: *Wissensorientierte Unternehmensführung*. Gabler Verlag, Wiesbaden, 1998
- [NoTa95] NONAKA, I.; TAKEUCHI, H.: *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press, New York, Oxford, 1995
- [Orwe49] ORWELL, G.: *Nineteen Eighty-Four*. Secker & Warburg, London, 1949
- [Öste87] ÖSTERLE, H.: *Erfolgsfaktor Informatik - Umsetzung der Informationstechnik in der Unternehmensführung*. In: IM, Fachzeitschrift für Information Management und Consulting, 38 (1987) 2, S. 24–31
- [OsWa97] OSTERLOH, M.; VON WARTBURG, I.: *Wissensmanagement*. In: Handelszeitung, (1997) 18
- [Ott92] OTT, H. J.: *Informationsmanagement*. In: WISU, das Wirtschaftsstudium, 21 (1992) 6, S. 476–482
- [O.V.96a] OHNE VERFASSER: *DIN Deutsches Institut für Normung e. V.: Geschäftsprozeßmodellierung und Workflow-Management. Forschungs- und Entwicklungsbedarf im Rahmen der Entwicklungsbegleitenden Normung (EBN)*. In: DIN Fachbericht 50, (1996)
- [O.V.96b] OHNE VERFASSER: *United States Government: Federal Standard 1037C - Glossary of Telecommunications Terms*, 1996
- [O.V.97] OHNE VERFASSER: *Institut für Lernende Organisation und Innovation (ILOI): Knowledge Management - Ein empirisch gestützter Leitfaden zum Management des Produktionsfaktors Wissen*. München, 1997
- [O.V.98a] OHNE VERFASSER: *ISDN-Telefon als Wanze - Lauschangriff auch bei aufgelegtem Hörer möglich?* In: Initiative Nachrichtenaufklärung, <http://www.kl-medien.de/ina/top.php?year=1998&title=SVNETi1UZWxlZm9uIGFscyBXYW56ZQ==>, (18. 11. 1998)
- [O.V.98b] OHNE VERFASSER: *Foebud e. V.: Big Brother Awards – Die Oscars für Datenkraken*. <http://big-brother-award.de>, 1998

- [O.V.99] OHNE VERFASSER: *mySAP.com Workplace Technology*. In: SAP Info, (1999)
- [O.V.00] OHNE VERFASSER: *Foebud e. V.: Was ist das Problem mit RFID?*
<http://www.foebud.org/texte/aktion/rfid/demo/htm/problem.html>, 2000
- [O.V.02] OHNE VERFASSER: *Consumers Against Supermarket Privacy Invasion and Numbering*.
<http://www.nocards.org>, 2002
- [O.V.04a] OHNE VERFASSER: *UbiSense Limited*. <http://www.ubisense.net>, 20. 12. 2004
- [O.V.04b] OHNE VERFASSER: *Mobile Positioning System*. <http://www.ericsson.com/mps>, 2004
- [O.V.04c] OHNE VERFASSER: *About the OSGi Service Platform*.
http://www.osgi.org/documents/osgi_technology/osgi-sp-overview.pdf, 22. 11. 2004
- [O.V.04d] OHNE VERFASSER: *The OSGi Service Platform - Dynamic services for networked devices*,
22. 11. 2004
- [O.V.04e] OHNE VERFASSER: *iButton - Touch the Future*.
Dallas Semiconductor Corp., <http://www.ibutton.com>, 23. 11. 2004
- [O.V.04f] OHNE VERFASSER: *Jobpilot.de: Berufsbild Informationsmanager*.
<http://www.jobpilot.de/content/journal/studium/beruf/informationsmanager.html>,
21.05.2004
- [O.V.04g] OHNE VERFASSER: *Xybernaut Solutions, Inc.: Case Study: Xybernaut Mobile Assistant:
Increasing the Efficiency of Aircraft Maintenance*, 2004
- [O.V.04h] OHNE VERFASSER: *Xybernaut Solutions, Inc.: Case Study: Xybernaut Mobile Assistant:
Increasing the Productivity of Personnel in the Field*, 2004
- [O.V.05a] OHNE VERFASSER: *BIB3R - Berlin's Beyond-3G Testbed and Serviceware Framework for
Advanced Mobile Solutions*. <http://www.bib3r.de>, 23.02.2005
- [O.V.05b] OHNE VERFASSER: *Sun Microsystems: JavaServer Pages Technology*.
<http://java.sun.com/products/jsp/>, 23.02.2005
- [PaHu03] PARK, J.; HUNTING, S.: *XML Topic Maps - Creating and Using Topic Maps for the Web*.
Boston, USA: Pearson Education, Inc., 2003
- [Pala97] PALASS, B.: *Der Schatz in den Köpfen*. In: Manager Magazine, (1997) 1, S. 112–121
- [Par⁺88] PARASURAMAN, A.; ZEITHAML, V. A.; BERRY, L. L.: *A Conceptual Model of Service
Quality and Its Implications for Future Research*. In: Journal of Marketing, (1988) 49,
S. 35–48
- [Pas⁺98] PASCOE, J.; RYAN, N.; MORSE, D.: *Human-Computer-Giraffe Interaction: HCI in the
Field*. In: JOHNSON, C. (Hrsg.): *Proceedings of the Workshop on Human Computer Inter-
action with Mobile Devices*, 1998

- [Pas⁺99] PASCOE, J.; RYAN, N.; MORSE, D.: *Issues in developing context-aware computing*. In: GELLERSEN, H.-W. (Hrsg.): *Handheld and Ubiquitous Computing, LNCS 1707*. Springer Verlag, Berlin, Heidelberg, New York, 1999
- [Pasc97] PASCOE, J.: *The Stick-e Note Architecture: Extending the Interface Beyond the User*. In: *International Conference on Intelligent User Interfaces, Orlando, Florida, 1997*
- [Pasc98] PASCOE, J.: *Adding Generic Contextual Capabilities to Wearable Computers*. In: *Proceedings of the Second International Symposium on Wearable Computing, Pittsburgh, 1998*, S. 92–99
- [PeMo01] PEPPER, S.; MOORE, G.: *XML Topic Maps (XTM) 1.0 - TopicMaps.Org Specification*. <http://www.topicmaps.org/xtml/>, 2001
- [Pent95] PENTLAND, A.: *Machine Understanding of Human Action*. In: *7th Int'l Forum on Frontier of Telecom Technology, Tokyo, Japan, 1995*
- [Pere97] PERENS, B.: *Open Source Definition, Version 1.9*. <http://www.opensource.org/docs/definition.php>, 1997
- [Pers99] PERSSON, C.: *Microsoft bestreitet Beihilfe zur Spionage*. In: Heise Online News, <http://www.heise.de/newsticker/meldung/6007>, (1999)
- [Pers00] PERSSON, C.: *Angeblich zigtausend Fehler in Windows 2000*. In: Heise Online News, <http://www.heise.de/newsticker/meldung/7979>, (12.02.2000)
- [Pers03] PERSSON, C.: *Teurer Wurm: Das hat SQLSlammer gekostet*. <http://www.heise.de/newsticker/meldung/34181>, 2003
- [Pesc02] PESCOVITZ, D.: *Helping Computers Help Themselves*. In: IEEE Spectrum Online, <http://www.spectrum.ieee.org/WEBONLY/publicfeature/sep02/auto.html>, (2002)
- [Pfoh03] PFOHL, H. C.: *Logistiksysteme*. Springer Verlag, Berlin, Heidelberg, New York, 2003
- [Pha⁺01] PHAM, T.-L.; SCHNEIDER, G.; GOOSE, S.; PIZANO, A.: *Composite Device Computing Environment: A Framework for Situated Interaction using Small Screen Devices*. In: *Personal and Ubiquitous Computing*, 5 (2001) 1
- [PiFr88] PICOT, A.; FRANCK, E.: *Die Planung der Unternehmensressource Information*. In: WISU, das Wirtschaftsstudium, (1988) 10, S. 544–549
- [PiFr93] PICOT, A.; FRANK, E.: *Aufgabenfelder eines Informationsmanagements*. In: WISU, das Wirtschaftsstudium, 22 (1993) 5, S. 433–437
- [PoBr99] POMEROL, J.; BREZILLON, P.: *Dynamics between contextual knowledge and proceduralized context*. In: ET AL., P. B. (Hrsg.): *Modeling and Using Context. Proc of CONTEXT'99, Trento, Italien, Springer-Verlag, Berlin, Heidelberg, 1999*

- [PoBr01] POMEROL, J.-C.; BREZILLON, P.: *About Some Relationships between Knowledge and Context*. In: *Proceedings of the International Conference on Modeling and Using Context (CONTEXT-01)*, 2001
- [Pon⁺01] PONNEKANTI, S. R.; LEE, B.; FOX, A.; HANRAHAN, P.; WINOGRAD, T.: *ICrafter: A Service Framework for Ubiquitous Computing Environments*. In: *Proceedings of the Second International Conference on Ubiquitous Computing, Atlanta*, 2001
- [Pove99] POVEY, D.: *Optimistic Security: A New Access Control Paradigm*. In: *Proceedings of the 1999 New Security Paradigms Workshop*, 1999
- [Powe03] POWER, R.: *Topic Maps for Context Management*. In: *Workshop on adaptive systems for ubiquitous computing, Dublin, Ireland*, 2003
- [Preß80] PRESSMAR, D. B.: *Abgrenzung der Berufsbilder in der Betriebsinformatik*. In: *Angewandte Informatik*, 22 (1980) 7, S. 275–279
- [Pri⁺00] PRIYANTHA, N. B.; CHAKARABORTY, A.; BALAKRISHNAN, H.: *The Cricket Location-Support System*. In: *Mobicom 2000, Boston*, 2000
- [PrKn98] PROBST, G.; KNAESE, B.: *Risikofaktor Wissen - Wie Banken sich vor Wissensverlusten schützen*. Gabler Verlag, Wiesbaden, 1998
- [Pro⁺98] PROBST, G.; RAUB, S.; ROMHARDT, K.: *Wissen managen - Wie Unternehmen ihre wertvollste Ressource optimal nutzen*. Gabler Verlag, Wiesbaden, 1998
- [Pupp88] PUPPE, F.: *Einführung in Expertensysteme*. 2. Auflage, Springer-Verlag, Berlin, Heidelberg, 1988
- [RaCa03] RANGANATHAN, A.; CAMPBELL, R.: *An infrastructure for context-awareness based on first order logic*. In: *Personal and Ubiquitous Computing*, (2003) 7, S. 353–364
- [Racc97] RACCAH, P. Y.: *Science, Language, and Situation*. In: *Proc. of the 2nd European conference on cognitive science, workshop on context (ECCS'97)*, 1997
- [Rak⁺01] RAKOTONIRAINY, A.; INDULSKA, J.; LOKE, S. W.; ZASLAVSKY, A.: *Middleware for Reactive Components: An Integrated Use of Context, Roles, and Event Based Coordination*. In: *Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms*, 2001, S. 77–98
- [RaSv95] RAYPORT, J. F.; SVIOKLA, J. J.: *Exploiting the Virtual ValueChain*. In: *Harvard Business Review*, 73 (1995) 6, S. 75–85
- [Reim98] REIMER, U.: *Building, Maintaining, and using Organisational Memories*. In: *ECAI'98 Workshop on Building, Maintaining, and Using Organisational Memories (OM-98)*, Brighton, England, 1998

- [Rek⁺98] REKIMOTO, J.; AYATSUKA, Y.; HAYASHI, K.: *Augment-able Reality: Situated Communication through Physical and Digital Spaces*. In: *ISWC*, 1998, S. 68–75
- [ReKr96] REHÄUSER, J.; KRCDMAR, H.: *Wissensmanagement im Unternehmen*. In: SCHREYÖGG, G. (Hrsg.); CONRAD, P. (Hrsg.): *Wissensmanagement*. de Gruyter, Berlin, 1996
- [ReLe00] REMUS, U.; LEHNER, F.: *The Role of Process-oriented Enterprise Modeling in Designing Process-oriented Knowledge Management Systems*. In: STAAB, S. (Hrsg.); O'LEARY, D. (Hrsg.): *AAAI Spring Symposium: Bringing Knowledge to Business Processes*, 2000
- [RoCI02] ROSENBAUM, J.; CLAESSENS, M.: *RTD Info - Magazine for European Research - Special Edition - The priorities of the Sixth Framework Programme 2002-2006*, Nov. 2002
- [Rom⁺02] ROMAN, M.; HESS, C. K.; CERQUEIRA, R.; RANGANATHAN, A.; CAMPBELL, R. H.; NAHRSTEDT, K.: *Gaia: a middleware infrastructure to enable active spaces*. In: *IEEE Pervasive Computing*, (2002) 10-12, S. 74–83
- [Rush02] RUSH, W.: *Immobots Take Control*. In: *Technology Review*, (2002), <http://www.technologyreview.com/articles/roush1202.asp>
- [SaMu03] SAHA, D.; MUKHERJEE, A.: *Pervasive Computing: a paradigm for the 21st century*. In: *IEEE Computer*, 36 (2003), März 3, S. 25–31
- [ScGe01] SCHMIDT, A.; GELLERSEN, H.-W.: *Modell, Architektur und Plattform für Informationssysteme mit Kontextbezug*. In: *Informatik Forschung und Entwicklung*, 16 (2001) 4, S. 213–224
- [ScGe03] SCHWOTZER, T.; GEIHS, K.: *Mobiles verteiltes Wissen*. In: *Datenbank-Spektrum*, (2003), 05, S. 30–39
- [Sch⁺93] SCHNEIDER-HUFSCHMIDT, M.; MALINOWSKI, U.; KUHME, T.: *Adaptive user interfaces: Principles and Practice*. Elsevier Science Inc., New York, USA, 1993
- [Sch⁺99a] SCHMIDT, A.; AIDOO, K.; TAKALUOMAI, A.; TUOMELAI, U.; LAERHOVEN, K. V.; DE VELDE, W. V.: *Advanced Interaction in Context*. In: GELLERSEN, H.-W. (Hrsg.): *International Symposium on Handheld and Ubiquitous Computing (HUC 99)*, Springer-Verlag, Berlin, Heidelberg, 1999, S. 89–101
- [Sch⁺99b] SCHMIDT, A.; BEIGL, M.; GELLERSEN, H.-W.: *There is more to Context than Location*. In: *Computers & Graphics Journal*, 23 (1999) 6, S. 893–902
- [Sch⁺00] SCHMIDT, A.; TAKALUOMA, A.; MÄNTYJÄRVI, J.: *Context-Aware Telephony over WAP*. In: *Personal Technologies*, 4 (2000) 4, S. 225–229
- [Sch⁺01a] SCHILIT, B. N.; TREVOR, J.; HILBERT, D.; KOH, T. K.: *m-Links: An Infrastructure for Very Small Internet Devices*. In: *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, Rome, Italy, 2001*, S. 122–131

- [Sch⁺01b] SCHMIDT, A.; STUHR, T.; GELLERSEN, H.-W.: *Context-Phonebook – Extending Mobile Phone Applications with Context*. In: *Third Mobile HCI Workshop*. University of Strathclyde, Glasgow, Scotland, 2001
- [Sch⁺01c] SCHWARZ, S.; ABECKER, A.; MAUS, H.; SINTEK, M.: *Anforderungen an die Workflow-Unterstützung für Wissensintensive Geschäftsprozesse*. In: MÜLLER, H. J. (Hrsg.); ABECKER, A. (Hrsg.); K.HINKELMANN (Hrsg.); MAUS, H. (Hrsg.): *Workshop 'Geschäftsprozessorientiertes Wissensmanagement' at WM'2001, Baden-Baden, Deutschland*, 2001
- [Sche78] SCHEER, A. W.: *Wirtschafts- und Betriebsinformatik*. Verlag Moderne Industrie, Landsberg, München, 1978
- [Sche88] SCHEER, A.-W.: *Wirtschaftsinformatik. Informationssysteme im Industriebetrieb*. Springer Verlag, Berlin, Heidelberg, 1988
- [Schi95] SCHILIT, B.: *System architecture for context-aware computing*, Columbia University, New York, Dissertation, 1995
- [Schm02] SCHMIDT, A.: *Ubiquitous Computing - Computing in Context*, Lancaster University, Dissertation, 2002
- [Schn02] SCHNEIDER, K.: *What to expect from Software Experience Exploitation*. In: *2nd Int. Conference on Knowledge Management (I-Know'02), Graz, Austria*, 2002, S. 149–156
- [Schö02] SCHÖN, E.: *XML und das Problem der Semantik*. In: *Java-Spektrum*, 6 (2002) 5, S. 22–29
- [Scho04] SCHOOP, E.: *Information in der Betriebswirtschaftslehre: ein neuer Produktionsfaktor?* In: KUHLEN, R. (Hrsg.); SEEGER, T. (Hrsg.); STRAUCH, D. (Hrsg.): *Grundlagen der praktischen Information und Dokumentation* Band 1. Saur Verlag, München, 2004
- [Schu01] SCHUMACHER, U.: *Kreative Köpfe brauchen Freiräume*. In: *Berliner Zeitung*, (2001)
- [Schu04] SCHULZKI-HADDOUTI, C.: *Europäische Informatik setzt auf "Ambient Intelligence"*. In: *Heise Online News*, 2004, <http://www.heise.de/newsticker/meldung/51363>
- [Schw01] SCHWOTZER, T.: *Wissensverteilung in allgegenwärtigen Systemen - Allgegenwärtiges Wissen*. In: *Herbsttreffen der GI Fachgruppe Betriebssysteme*, 2001, <http://ivs.tu-berlin.de/thsc/allWissen.pdf>
- [Schw02] SCHWOTZER, T.: *Context Driven Spontaneous Knowledge Exchange*. In: *Proceedings of the 1st German Workshop on Experience Management 2002 (GWEM02), Berlin, Germany*, 2002
- [Schw03] SCHWEIKER, A.: *Wie verändern mobile Technologien Geschäftsprozesse in der Ver- und Entsorgungswirtschaft?* In: *Proceedings der WI 2003*, Physica-Verlag, Heidelberg, 2003

- [Schw04] SCHWOTZER, T.: *Modelling Distributed Knowledge Management Systemes with Topic Maps*. In: *Proc. of the 4th International Conference on Knowledge Management (I-Know '04)*, Graz, Austria, 2004
- [Scri02] SCRIMSHAW, P.: *Total cost of ownership: A review of the literature*. In: *ICT in Schools Research and Evaluation Series*, (2002)
- [ScTh94] SCHILIT, B. N.; THEIMER, M. M.: *Disseminating Active Map Information to Mobile Hosts*. In: *IEEE Network*, IEEE Computer Society, 8 (1994) 5, S. 22–32
- [Shmu87] SHMUELI, O.: *Decidability and expressiveness aspects of logic queries*. In: *Proceedings of the Sixth ACM SICACT-SIGMOD-SIGART Symposium on the Principles of Database Systems*, San Diego, CA, 1987
- [SiDe93] SIMON, J.; DENNIS, G.: *Methods of Tracking Intersystem Configuration Management Dependencies*. In: *Journal of Systems Management*, (1993)
- [Sied99] SIEDENTOPF, J.: *Das Rekonstruktionsparadigma in der reformulierten starken KI-These*. In: BECKER, J. (Hrsg.); KÖNIG, W. (Hrsg.); SCHÜTTE, R. (Hrsg.); WENDT, O. (Hrsg.); ZELEWSKI, S. (Hrsg.): *Wirtschaftsinformatik und Wissenschaftstheorie. Bestandsaufnahme und Perspektiven*. Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, Wiesbaden, 1999, S. 393–415
- [SiSt01a] SINGLETARY, B.; STARNER, T.: *Learning Visual Models of Social Engagement*. In: *Intl. Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Realtime Systems, Vancouver, Canada, 2001*, S. 141–148
- [SiSt01b] SINGLETARY, B.; STARNER, T.: *Symbiotic interfaces for wearable face recognition*. In: *Human Computer Interaction International (HCII), New Orleans, 2001*
- [Sloa92] SLOANE, L.: *Orwellian Dream Come True: A Badge That Pinpoints You*. In: *New York Times*, (1992), S. 14
- [Smit75] SMITH, A.: *Untersuchung über die Natur und die Ursachen des Reichtums der Nationen*, 1775
- [Smy⁺02] SMYTH, B.; BRADLEY, K.; RAFTER, R.: *Personalization Techniques for Online Recruitment Services*. In: *Communications of the ACM*, 45 (2002) 5, S. 39–40
- [Sonn99] SONNTAG, R.: *Organisation aktiver Umwelthandbücher - Ein Beitrag zum prozeßorientierten Dokumentenmanagement*, TU-Dresden, Dissertation, 1999
- [Spoh99] SPOHRER, J.: *Information in Places*. In: *IBM Systems Journal - Pervasive Computing*, 38 (1999) 4
- [Spr⁺01] SPRIESTERSBACH, A.; VOGLER, H.; LEHMANN, F.; ZIEGERT, T.: *Integrating Context Information into Enterprise Applications for the Mobile Workforce*. In: *Proceedings of MobiCom 2001*, 2001

- [Spri04] SPRINGER, T.: *Ein komponentenbasiertes Meta-Modell kontextabhängiger Adaptionsgraphen für mobile und ubiquitäre Anwendungen*, Technische Universität Dresden, Dissertation, 2004
- [Srin97] SRINIVAS, K.: *How is context represented in implicit and explicit memory*. In: *Proc. of the 2nd European Conference on Cognitive Science, Workshop on Context (ECCS'97)*, 1997
- [Sta⁺97] STARNER, T.; MANN, S.; RHODES, B.; LEVINE, J.; HEALEY, J.; KIRSCH, D.; PICARD, R.; PENTLAND, A.: *Augmented Reality Through Wearable Computing*. In: *Presence*, 6 (1997) 4
- [Sta⁺98] STARNER, T.; WEAVER, J.; PENTLAND, A.: *A Wearable Computing Based American Sign Language Recognizer*. In: MITTAL, V. (Hrsg.); YANCO, H. (Hrsg.); ARONIS, J. (Hrsg.): *Assistive Technology and Artificial Intelligence* Band 4. Springer Verlag, Berlin, Heidelberg, New York, 1998
- [Sta⁺00] STARNER, T.; AUXIER, J.; ASHBROOK, D.; GANDY, M.: *The Gesture Pendant: A Self-luminating, Wearable, Infrared Computer Vision System for Home Automation Control and Medical Monitoring*. In: *International Symposium on Wearable Computers, Atlanta, 2000*, S. 87–94
- [Sta⁺03] STARNER, T.; LEIBE, B.; MINNEN, D.; WESTEYN, T.; HURST, A.; WEEKS, J.: *Integration of Wireless Gesture Tracking, Object Tracking, and 3D Reconstruction in the Perceptive Workbench*. In: *Machine Vision and Applications*, (2003)
- [Star99a] STARNER, T.: *The Everyday Use of Wearable Computers*. In: *Interaction Homme-Machine (IHM'99), Montpellier, France, 1999*
- [Star99b] STARNER, T.: *Wearable Computing and Contextual Awareness*., Massachusetts Institute of Technology, Dissertation, Juni 1999
- [Star02a] STARNER, T.: *Attention, Memory, and Wearable Interfaces*. In: *IEEE Pervasive Computing*, 4 (2002), Oktober 1, S. 88–91
- [Star02b] STARNER, T.: *The Role of Speech Input in Wearable Computing*. In: *IEEE Pervasive Computing*, 1 (2002), Juli 3, S. 89–93
- [StHa99] STAHLKNECHT, P.; HASENKAMP, U.: *Einführung in die Wirtschaftsinformatik*. Springer Verlag, Berlin, Heidelberg, 1999
- [Sti⁺98] STILLMAN, S.; TANAWONGSUWAN, R.; ESSA, I.: *A system for tracking and recognizing multiple people with multiple cameras*. Georgia Institute of Technology, Forschungsbericht, 1998
- [Stoc00] STOCK, W. G.: *Informationswirtschaft: Management externen Wissens*. Oldenbourg Wissenschaftsverlag GmbH, München, Wien, 2000

- [Str⁺98] STREITZ, N.; HARTKOPF, V.; ISHII, H.; KAPLAN, S.; MORAN, T. P.: *Cooperative Buildings - Integrating Information, Organization and Architecture*. In: *Proceedings of the First International Workshop on Cooperative Buildings (CoBuild 98)*, 1998, S. 4–21
- [Svei98] SVEIBY, K. E.: *Das Wissenskapital – das unentdeckte Vermögen*. Verlag Moderne Industrie, Landsberg, 1998
- [SzEi83] SZYPERSKI, N.; EICHENRÖDER, G.: *Information-Resource-Management*. In: KAY, R. (Hrsg.): *Management betrieblicher Informationsverarbeitung*. Oldenbourg, München, 1983
- [Tolk54] TOLKIEN, J.: *The Lord of the Rings*. George Allen & Unwin Ltd., London, 1954
- [Trae94] TRAEGER, D.: *Einführung in die Fuzzy Logik*. Teubner, Stuttgart, 1994
- [Tulv83] TULVING, E.: *Elements of episodic memory*. Oxford University Press, 1983
- [Turn92] TURNER, R. M.: *A View of Diagnostic Reasoning as a Memory-Directed Task*. In: *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, 1992
- [TuTu91] TURNER, E. H.; TURNER, R. M.: *A Schema-based Approach to Cooperative Behavior*. In: *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, 1991, S. 928–932
- [VaBo93] VANDERDONCKT, J.; BODART, F.: *Encapsulating Knowledge for Intelligent Automatic Interaction Objects Selection*. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1993, S. 424–429
- [Vett89] VETTER, M.: *Das Jahrhundertproblem der Informatik*. In: MÜLLER-ETTRICH (Hrsg.): *Effektives Datenbankdesign*. Köln, 1989, S. 11–31
- [Vol⁺99] VOLZ, S.; FRITSCH, D.; D., K.; LEONHARDI, A.; SCHÜTZNER, J.: *NEXUS - Spatial Model Servers for Location-Aware Applications on the Basis of ArcView*. In: *Proceedings of the 14th ESRI European User Conference*, 1999
- [Vroo64a] VROOM, V. H.: *Expectancy Theory*. John Wiley and Sons, 1964
- [Vroo64b] VROOM, V. H.: *Work and Motivation*. Jossey-Bass, 1964
- [WaAk00] WANG, W.; AKYILDIZ, I.: *Intersystem Location Update and Paging Schemes for Multitier Wireless Networks*. In: *Mobicom 2000, Boston*, 2000
- [Wan⁺92] WANT, R.; HOPPER, A.; FALCAO, V.; GIBBONS, J.: *The Active Badge Location System*. In: *ACM Transactions on Information Systems*, 10 (1992) 1, S. 91–102
- [Wan⁺95] WANG, R.; REDDY, M. P.; KON, H.: *Towards Quality Data: An attribute-based approach*. In: *Decision Support Systems*, (1995) 13, S. 349–372

- [Wan⁺96] WANT, R.; SCHILIT, B. N.; ADAMS, N. I.; GOLD, R.; PETERSEN, K.; GOLDBERG, D.; ELLIS, J. R.; WEISER, M.: *Overview of the PARCTAB Ubiquitous Computing Experiment*. In: *Mobile Computing*, (1996)
- [WaNo86] WALLIS, J. J.; NORTH, D. C.: *Measuring the Cransaction Sector in the American Ecoromy, 1870-1970*. In: ENGERMAN, S. L. (Hrsg.); GALLMAN, R. E. (Hrsg.): *Long-Term Factors in American Economic Growth*. Chicago/London, 1986
- [Weis91] WEISER, M.: *The computer for the twenty-first century*. In: *Scientific American*, 265 (1991) 3, S. 86–93
- [WeMa01] WENZEL, C.; MAUS, H.: *Leveraging corporate context within knowledge-based document analysis and understanding*. In: *Int. Journal on Document Analysis and Recognition, Special Issue on Document Analysis for Office Systems*, 2001
- [Wenz98] WENZEL, C.: *Integrating information extraction into workflow management systems*. In: *Natural Language and Information Systems Workshop (NLIS/DEXA 98)*, Wien, Österreich, 1998
- [Wer⁺01] WERLE, P.; KILANDER, F.; JONSSON, M.; LÖNNQVIST, P.; JANSSON, C.: *A Ubiquitous Service Environment with Active Documents for teamwork support*. In: *Proceedings of the International Conference on Ubiquitous Computing (UbiComp 2001)*, Atlanta, USA, 2001
- [West67] WESTIN, A. F.: *Privacy and Freedom*. Atheneum, New York, 1967
- [Will96] WILLIAMS, B. C.: *Model-based Autonomous Systems in the New Millennium*. In: *Proceedings of AI Planning Systems*, <http://ic.arc.nasa.gov/ic/projects/mba/papers/aips96.ps>, (1996)
- [WiNa96] WILLIAMS, B. C.; NAYAK, P. P.: *Immobile Robots: AI in the New Millennium*. In: *AI Magazine*, <http://ic.arc.nasa.gov/ic/projects/mba/papers/aimag96.ps>, (1996)
- [Wis⁺98] WISNESKI, G.; ISHII, H.; DAHLEY, A.; GORBET, M.; BRAVE, S.; ULLMER, B.; YARIN, P.: *Ambient Display: Turning Architectural Space into an Interface between People and Digital Information*. In: *Proceedings of the First International Workshop on Cooperative Buildings (CoBuild'98)*, Springer Verlag, Berlin, Heidelberg, New York, 1998
- [Wol⁺99] WOLFF, B.; FUCHS-KITTOWSKI, K.; KLISCHEWSKI, R.; MÖLLER, A.; ROLF, A.: *Organisationstheorie als Fenster zur Wirklichkeit*. In: BECKER, J. (Hrsg.); KÖNIG, W. (Hrsg.); SCHÜTTE, R. (Hrsg.); WENDT, O. (Hrsg.); ZELEWSKI, S. (Hrsg.): *Wirtschaftsinformatik und Wissenschaftstheorie. Bestandsaufnahme und Perspektiven*. Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, Wiesbaden, 1999
- [Wom⁺91] WOMACK, J. P.; JONES, D. T.; ROOS, D.: *The Machine That Changed the World: The Story of Lean Production*. Perennial, 1991

- [Wre⁺95] WREN, C.; DARRELL, T.; STARNER, T.; JOHNSTON, M.; RUSSELL, K.; AZARBAYEJANI, A.; PENTLAND, A.: *Pfinder: A Real-Time System for Tracking People*. In: BOVE, M. (Hrsg.): *SPIE Conference on Real-Time Vision, Philadelphia, USA, 1995*
- [Wrig98] WRIGHT, S.: *Entwicklungen in der Überwachungstechnologie*. In: Telepolis, <http://www.heise.de/tp/r4/artikel/6/6246/s2.html>, (23. 06. 1998)
- [WrSp96] WREN, C. R.; SPARACINO, F.: *Perceptive Spaces for Performance and Entertainment: Untethered Interaction using Computer Vision and Audition*. In: *Applied Artificial Intelligence (AAI) Journal*, (1996)
- [Wyc⁺98] WYCKOFF, P.; MCLAUFHRY, S. W.; LEHMAN, T. J.; FORD, D. A.: *T Spaces*. In: *IBM Systems Journal - Pervasive Computing*, (1998), S. 454–474
- [Zade73] ZADEH, L. A.: *Outline of a New Approach to the Analysis of Complex Systems*. In: *IEEE Transactions on System, Man and Cybernetics*, 3 (1973), Jan. 1, S. 28–44
- [ZaRü89] ZAHN, E.; RÜTTLER, M.: *Informationsmanagement. Eine strategische Antwort auf kritische Herausforderungen der Unternehmenswelt*. In: *Controlling*, 1 (1989) 1, S. 34–43
- [ZuSc94] ZUCKER, B.; SCHMITZ, C.: *Wissen nutzen statt verspielen*. In: *Gabler Magazine*, (1994) 11/12, S. 62–65

Anhang A Detaillierte Bewertungen verwandter Arbeiten

Anhang A.1 Entity-Relationship-Modell

Tabelle 27: Bewertung des Entity-Relationship-Modells hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	4	ERM enthalten stark strukturierte Daten, allerdings nicht kontextspezifisch formalisiert. Es fehlt eine hierarchische Formalisierungsmöglichkeit.
1b	Modularität	1	Daten können zwar in verschiedenen Tabellen gespeichert werden, doch kann diese Unterteilung nicht willkürlich vorgenommen und nicht beliebig verfeinert werden.
1c	Verteilte Darstellung und Speicherung	1	Relationale Inhalte lassen sich relativ gut replizieren, allerdings stellt das RDM dazu keine methodische Unterstützung bereit. Inhalte in verschiedenen RDB bzw. RDBS lassen sich nicht direkt adressieren. Repliken beziehen sich nur auf Daten, nicht die diese Daten auswertenden Anwendungen, die separat repliziert werden müssten.
1d	Unabhängigkeit von Programmiersprachen	4	RDB stellen mit SQL eine standardisierte, datenorientierte, programmiersprachenunabhängige Schnittstelle zur Außenwelt zur Verfügung. Code kann darin nicht dargestellt werden.
1e	Flexible Adressierungsmöglichkeiten	4	Zwar können Assoziationen zwischen Realweltobjektmodellen dargestellt werden, jedoch können diese Verbindungen nicht dynamisch berechnet werden.
2a	Separate Strukturierung von Datendimensionen	3	Es ist zwar möglich, ein Datenmodell zu erschaffen, das mehrere Aspekte der Realwelt widerspiegelt, jedoch wird dies nicht im besonderen unterstützt und ist daher sehr unflexibel. Separate Datendimensionen können im RDM zwar strukturiert, aber nicht in der benötigten flexiblen Weise miteinander kombiniert werden. Auch können separate Ontologien modellseitig nicht miteinander kombiniert werden.
2b	Kombination separat strukturierter Datendimensionen	1	Separate Ontologien können nicht ohne weiteres miteinander kombiniert werden. Dies macht weitergehende formale Spezifikationen, wie sie u. a. in [Kosk91] beschrieben werden, notwendig.
3a	Dynamische Elemente	2	Die Dynamik von Datenfeldern kann im ERM nicht modelliert werden. Jedoch können über Trigger Veränderungen in der Datenbasis von externen Anwendungen abonniert werden.
3b	Generalisierende Ontologien	1	Zur Laufzeit lassen sich lediglich neue Attribute zu bestehenden Objekttypen hinzufügen, durch die fehlenden Klassenhierarchien ist es nicht möglich, auch für bestehende Anwendungen verständliche neue Objekttypen zu definieren.
3c	Darstellung der Historie von Kontextvariablen	2	Zur Darstellung der Historie von relationalen Daten gibt es kein standardisiertes Vorgehen. Dies wäre nur durch proprietäre Erweiterungen des SQL-Standards möglich.
4a	Datenqualitätsdeskriptoren	2	RDB dienen vorwiegend zur Speicherung sicherer Daten, die abgesehen von offensichtlichen Konsistenzproblemen nicht vom RDBMS in Frage zu stellen sind. Attribute zur Datenqualität können zum Datenschema hinzugefügt werden, allerdings wäre dies proprietärer Natur und könnte nicht von allen beteiligten Anwendungen konsistent interpretiert und nur sehr eingeschränkt von dem RDBMS zur Administration der Informationen berücksichtigt werden.
4b	Datentypspez. Qualitätsdeskriptoren	1	Da im ERM nur sehr grobe Datentypen unterschieden werden können, ist eine datentypspezifische Gestaltung von Qualitätsattributen nicht ohne weiteres möglich.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
5a	Datenschutzdeskriptoren	2	Für Datenschutzattribute gilt im Prinzip das selbe wie für die bereits besprochene Datenqualität.
5b	Darstellung von Akteuren	1	Die auf der Darstellung von Akteuren beruhende dynamische Bestimmung von Zugriffsrechten ist im ERM nicht vorgesehen und kann ohne substanzielle Erweiterungen dieses Standards nicht realisiert werden.
5c	Vertrauensstufen	1	Weder können im ERM Vertrauensstufen dargestellt werden, noch unterschiedliche Repräsentationen desselben Datums für verschiedene Vertrauensstufen.
6a	Realweltobjekt-bezogene Struktur	5	Im ERM werden Realweltobjektmodelle samt dazugehörigen Attributen dargestellt. Attribute können bei Kenntnis des entsprechenden Realweltobjektes in gesammelter Form zugegriffen werden.
6b	Eindeutige Realweltobjektmodelle	2	Entities werden nur tabellenintern eindeutig gekennzeichnet. Der SQL-Standard sieht keine Methode vor, um für einen Datensatz eine global gültige ID festzulegen. Die Zuordnung von Realweltobjekten zu Realweltobjektmodellen wird durch das ERM in keiner Weise festgelegt.
6c	Explizite Ontologie	3	Relationale Datenbanken verwenden ein explizites Datenschema, das u. a. durch das ERM und seinen Abkömmlingen definiert werden kann. Dabei können jedoch nicht alle Anforderungen an eine Ontologie dargestellt werden, etwa die Darstellung von Ober- und Unterkategorien von Entities.
6d	Darstellung von Kontextquellen	1	Das ERM kann keine externen Datenquellen darstellen, um deren Inhalte in das eigene Modell zu integrieren.
6e	Inhaltliche Darstellung von Kontextvariablen	3	Inhalte von externen Datenquellen können im Kontextmodell in einer verarbeitbaren Form dargestellt werden, jedoch kann dieser Vorgang nicht automatisiert erfolgen.
6f	Verbindung vom Kontextmodell zu den Originaldaten	1	Mit den durch das ERM gegebenen Mitteln ist die Adressierung von externen Daten, um sie gleichwertig mit internen Daten verwenden zu können, nicht möglich.
6g	Fusion von Kontextvariablen	1	Synonyme Attribute aus verschiedenen externen Kontextquellen können innerhalb eines ERM weder identifiziert noch inhaltlich kombiniert werden.
6h	Integration von Methodik in das Datenmodell	1	Abgesehen von Triggern kann Code im ERM oder RDM nicht innerhalb der Datenbasis dargestellt werden.
6i	Darstellung des Datenpfades	3	Die Aktualisierung abhängiger Variablen könnte über Trigger realisiert werden, allerdings können dadurch nicht die in die Neuberechnung einfließenden Inhalte modelliert werden. Der Trigger-Ansatz entspricht dem bereits in Kap. 4.3.1 („Externe Agenten“, S. 108) kritisierten Ansatz von <i>externen Agenten</i> .
6j	Einschränkung der Sichtbarkeit von Konzepten	1	Sämtliche Elemente eines ERM sind überall sicht- und zugreifbar, was mit den gegebenen Mitteln auch nicht eingeschränkt werden kann.
6k	Synonyme Benennung von Topics	2	Entities in einer RDB werden zunächst nur über ihre interne ID adressiert. Eine separate Darstellung synonymen Namen für Entities wäre sehr einfach zu realisieren, aufgrund einer fehlenden Verankerung im Standard jedoch nicht allen Anwendungen geläufig.
7a	Darstellung verschiedener Abstraktionsstufen	1	Im ERM können weder verschiedene Abstraktionsstufen ein und derselben Information, noch Abhängigkeiten zwischen verschiedenen abstrakten Repräsentationen hergestellt werden.
7b	Endnutzertauglich Programmiermethodiken	4	Zur Benutzung von in einer RDB dargestellten Kontextinformationen sind außer dem geläufigen, relativ einfach zu erlernenden und flexibel benutzbaren SQL keinerlei Fachkenntnisse nötig. Allerdings sind die dabei verwendeten relationalen, stark normalisierten Datenschemata nicht immer intuitiv verständlich.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
7c	Problemspezifische Sichtweisen	4	Mit Views können anwendungsspezifische, subjektive Sichtweisen auf beliebige Teile der Datenbasis generiert werden. Allerdings können Views nicht immer auch zum Schreiben verwendet werden. Auch können zwar die Namen, nicht jedoch die betreffenden Inhalte etwa hinsichtlich ihrer Abstraktionsstufe oder Darstellungsform verändert werden.

Anhang A.2 Objektorientiertes Datenmodell

Tabelle 28: Bewertung des objektorientierten Datenmodells hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	4	Das OODM erlaubt die explizite, hierarchische Definition von Objektstrukturen, allerdings nicht von Assoziationen.
1b	Modularität	5	Modularität ist einer der Vorteile objektorientierter Datenstrukturen.
1c	Verteilte Darstellung und Speicherung	4	Die in OODB gespeicherten Objektstrukturen lassen sich gut auf verschiedene Endgeräte replizieren, allerdings nur wenn auch die entsprechenden Codestrukturen auf dem betreffenden Endgerät lauffähig sind.
1d	Unabhängigkeit von Programmiersprachen	1	Objektorientierte Datenbanken sind nur begrenzt unabhängig von Programmiersprachen, sie sind i. d. R. auf eine spezielle Anwendung zugeschnitten und in deren Anwendungscode integriert.
1e	Flexible Adressierungsmöglichkeiten	5	Objekte können beliebig komplexe Assoziationen untereinander darstellen, die auch zur Laufzeit bestimmt werden können.
2a	Separate Strukturierung von Datendimensionen	5	Falls multiple Vererbung möglich ist, lassen sich verschiedene Kontextdimensionen separat modellieren, allerdings nicht in einer kontextbezogen formalisierten Form. Dabei werden jedoch Implementations- und Kontextaspekte vermengt.
2b	Kombination separat strukturierter Datendimensionen	5	Bei der Möglichkeit zu multipler Vererbung können konkrete Objektklassen aus mehreren, separat modellierten Superklassen abgeleitet werden.
3a	Dynamische Elemente	4	In OODB können wenig veränderliche Daten als statische Attribute und schnell veränderliche Daten durch Funktionen dargestellt werden. Weitergehende Unterscheidungen etwa hinsichtlich Caching sind nicht möglich, können aber manuell programmiert werden.
3b	Generalisierende Ontologien	5	Die bei OODB verwendeten Datenschemata sind gemäß einer hierarchischen Ontologie organisiert, die auch Anwendungen bewusst ist.
3c	Darstellung der Historie von Kontextvariablen	4	Der OODB-Standard sieht eine Versionierung der Inhalte vor, was sich bedingt zur Historisierung von Kontextinformationen eignet.
4	Datenqualität	2	Der OODB-Standard sieht keine Beschreibung der Datenqualität vor, jedoch wäre dies einfach über entsprechende Attribute lösbar, jedoch proprietär.
5	Schutz der Privatsphäre	2	Der OODB-Standard sieht keine Beschreibung von Datenschutzaspekten vor, jedoch wäre dies einfach über entsprechende Attribute lösbar, jedoch proprietär.
6a	Realweltobjektbezogene Struktur	5	Beim OODM werden Kontextvariablen als Attribute von separat modellierten Objektstrukturen dargestellt und zusammen mit diesen gespeichert.
6b	Eineindeutige Realweltobjektmodelle	2	Jedes Objekt wird über eine innerhalb der OODB eineindeutige ID angesprochen, die jedoch nicht global eindeutig ist. Mechanismen zur global eindeutig Identifizierung von Objekten wären im OODM einfach realisierbar, sind jedoch nicht standardisiert. Ebenfalls nicht standardisiert ist der Zusammenhang zwischen Objekt-ID und dem entsprechenden Realweltobjekt.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6c	Explizite Ontologie	4	Das OODM basiert auf einer expliziten, als Ontologie interpretierbaren Klassendefinition, welche die wichtigsten ontologischen Funktionen erfüllt. Allerdings können Objekte i. d. R. nur einer Ontologie zugeordnet werden.
6d	Darstellung von Kontextquellen	2	Die explizite Darstellung externer Datenquellen ist im objektorientierten Datenmodell nicht standardisiert, könnte aber durch entsprechende Objekte relativ einfach realisiert werden.
6e	Inhaltliche Darstellung von Kontextvariablen	4	Externe Informationen können wertmäßig im OODM dargestellt werden, was durch die Einbettung von Zugriffsroutinen automatisiert werden könnte, jedoch ist dieser Vorgang nicht standardisiert.
6f	Verbindung vom Kontextmodell zu den Originaldaten	2	Externe Daten können im OODM transparent durch beliebig komplexe Zugriffsmethoden eingebunden werden. Allerdings gibt es hierfür keine festgelegten Standards.
6g	Fusion von Kontextvariablen	2	Da externe Informationen nicht dargestellt werden können, ist auch die Kombination synonyme externer Informationen nicht möglich.
6h	Integration von Methodik in das Datenmodell	5	Sämtliche geforderten Funktionalitäten werden durch das OODB bereitgestellt.
6i	Darstellung des Datenpfades	4	Sämtliche im Laufe der Berechnung von höherwertigen Kontextinformationen notwendigen Informationen sind im objektorientierten Datenmodell enthalten, allerdings innerhalb von Codesegmenten und damit nicht explizit genug. Darüber hinaus sind die Rollen der beteiligten Information nicht semantisch definiert.
6j	Einschränkung der Sichtbarkeit von Konzepten	3	Die Sichtbarkeit von Attributen kann im OODM in 3 Granularitätsstufen angegeben werden. Private Attribute sind nur innerhalb ihres Objektes sichtbar, geschützte Attribute nur im jeweiligen Modul und nur als öffentlich deklarierte Attribute sind global sichtbar. Dies bezieht sich allerdings nicht auf die Objekte selbst, die immer global sichtbar sind. Zudem sind diese Grenzen nicht willkürlich festleg- und nur begrenzt veränderbar.
6k	Synonyme Benennung von Topics	2	Der OODB-Standard selbst sieht keine synonyme Benennung von Objekten bzw. Attributen vor, jedoch kann dies durch proprietäre Methoden nachgebildet werden.
7a	Darstellung verschiedener Abstraktionsstufen	1	Das OODBM kennt keine Abstraktionsebenen. Entsprechend können keine Detailinformationen zu verschiedenen Abstraktionsebenen. Diese Funktionalität kann nur durch fundamentale Designänderungen realisiert werden.
7b	Endnutzertaugliche Programmiermethoden	1	Die Veränderung einer OODB durch einen relativ unbedarften Endnutzer ist nicht ohne weiteres möglich, da die betreffenden Datenstrukturen im jeweiligen Anwendungscode eingebettet sind.
7c	Problemspezifisch Sichtweisen	1	OODB können prinzipiell beliebig vernetzte Objektstrukturen und damit mehrere subjektive Sichtweisen auf die Datenbasis erzeugen, allerdings sind solche Unternehmungen nicht standardisiert, d. h. es fehlt eine standardisierte Unterteilung in verschiedene Sichtweisen sowie eine methodische Unterstützung bei der Unterstützung einer anderen Ontologie.

Anhang A.3 XML

Tabelle 29: Bewertung der Extensible Markup Language hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	5	Sämtliche Modellelemente können mittels DTD oder XML-Schema formal definiert werden. Nur XML-Schema ist zur hierarchischen Definition der verwendeten Datentypen mittels Vererbung in der Lage.
1b	Modularität	5	XML erlaubt die flexible Unterteilung und Kombination von Modellen mittels einzelner XML-Dateien und stellt mit <i>XPath</i> und <i>XPointer</i> Methoden zur Adressierung von Informationen in anderen XML-Dateien zur Verfügung.
1c	Verteilte Darstellung und Speicherung	1	Da keine Codesegmente in XML dargestellt werden können, sind auf XML basierende Kontextmodelle nur begrenzt replikationsfähig.
1d	Unabhängigkeit von Programmiersprachen	5	XML ist aufgrund seines datenorientierten Charakters vollständig unabhängig von den verwendeten Programmiersprachen.
1e	Flexible Adressierungsmöglichkeiten	3	XML erlaubt die Darstellung von Assoziationen zwischen Datenstrukturen, allerdings nur in einer statischen Form.
2a	Separate Strukturierung von Datendimensionen	5	Separate Datendimensionen können durch getrennte XML-Schemata modelliert werden.
2b	Kombination separat strukturierter Datendimensionen	5	Separate XML-Schema können mittels <i>Namespaces</i> kombiniert werden.
3a	Dynamische Elemente	1	XML geht nur von statischen Inhalten aus.
3b	Generalisierende Ontologien	5	Die von XML verwendeten Ontologien können zur Laufzeit erweitert werden.
3c	Darstellung der Historie von Kontextvariablen	2	XML selbst stellt keine Mittel bereit, um vergangene Werte von XML-Strukturen darzustellen. Solche Historien wären einfach realisierbar, allerdings nur proprietär.
4a	Datenqualitätsdeskriptoren	2	Der XML-Standard sieht keine Mechanismen zur Darstellung der Datenqualität vor.
4b	Datentypspez. Qualitätsdeskriptoren	2	XML-Schema erlauben neben der Nutzung vordefinierter atomarer Datentypen die Definition eigener, beliebig komplexer Attribute. Diese können zur Definition verschiedener Kontextvariablentypen verwendet werden. Die Zuordnung von datentypspezifischen Qualitätsmetriken muß jedoch durch modellinterne Mechanismen und damit nichtstandardisiert erfolgen.
5a	Datenschutzdeskriptoren	2	Datenschutzaspekte können zwar gut in XML dargestellt werden, sind aber nicht im speziellen durch den Standard formalisiert.
5b	Darstellung von Akteuren	2	Realweltakteure könnten gut in XML dargestellt werden, dies ist aber nicht im speziellen durch den Standard formalisiert.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
5c	Vertrauensstufen	2	Vertrauensstufen könnten gut in XML dargestellt werden, dies ist aber nicht im speziellen durch den Standard formalisiert.
6a	Realweltobjekt-bezogene Struktur	5	In XML können Realweltobjektmodelle durch <i>Elemente</i> und zu diesen Realweltobjekten gehörende Kontextvariablen durch <i>Attribute</i> oder eingebettete Elemente dargestellt werden.
6b	Eineindeutige Realweltobjektmodelle	2	XML per se macht keine Vorschriften hinsichtlich der Eineindeutigkeit seiner Inhalte. Realweltobjekte und ihre Modelle können in XML eindeutig per URI adressiert werden.
6c	Explizite Ontologie	5	XML-formatierte Informationen basieren i. d. R. auf expliziten Ontologien, die als DTD, XML-Schema oder Relax formatiert sein können (vgl. [FaWa04, CIMa01]). Dabei können verschiedene Ontologien miteinander per <i>Namespace</i> kombiniert werden.
6d	Darstellung von Kontextquellen	2	Externe Datenquellen könnten durch XML-Datenstrukturen modelliert werden, allerdings ist dies nicht formalisiert.
6e	Inhaltliche Darstellung von Kontextvariablen	3	Externe Inhalte können in XML-Dokumenten dargestellt werden, jedoch fehlen Mechanismen zur Automatisierung vollständig.
6f	Verbindung vom Kontextmodell zu den Originaldaten	4	Zumindest XML-basierte Datenstrukturen aus externen Datenbeständen können mittels XPointer adressiert werden. Für andere Datenstrukturen stellt XML keine Adressierungsmöglichkeiten bereit.
6g	Fusion von Kontextvariablen	1	Da keine Operationen auf Inhalten dargestellt werden können, können die Inhalte von XML-Datenbanken mit den durch XML bereitgestellten Mitteln nicht miteinander kombiniert werden.
6h	Integration von Methodik in das Datenmodell	1	XML ist rein datenorientiert, d.h. methodische Elemente können nicht innerhalb von XML-Dokumenten dargestellt werden.
6i	Darstellung des Datenpfades	1	Da in XML keine Transformationen von Inhalten dargestellt werden können, ist es auch nicht möglich, den Transformationspfad darzustellen.
6j	Einschränkung der Sichtbarkeit von Konzepten	1	Die Sichtbarkeit von Elementen kann in XML nicht auf bestimmte frei definierbare Bereiche eingeschränkt werden.
6k	Synonyme Benennung von Topics	2	Synonyme Objektnamen können in XML-Schema durch eine <i>Substitution</i> dargestellt werden, wobei jedoch die Semantik, dass es sich um synonyme Namen handeln kann, nicht dargestellt werden kann.
7a	Darstellung verschiedener Abstraktionsstufen	2	XML stellt keine formalisierten Mittel zur Verfügung, um verschiedene Abstraktionsstufen einer Datenbasis darzustellen. Aufgrund der Flexibilität von XML wäre so etwas jedoch einfach zu realisieren.
7b	Endnutzertauglich Programmiermethodiken	3	XML stellt einen hierarchischen Baum von Informationen dar, der prinzipiell endnutzertauglich visualisiert werden kann. Dies bezieht sich jedoch nur auf den Datenteil der kontextbezogenen Wissensbasis.
7c	Problemspezifische Sichtweisen	2	Zwar kann XML flexibel mit verschiedenen Ontologien umgehen, jedoch ist die hier geforderte transparente Darstellung von Informationen in mehreren Ontologien nicht ohne Erweiterung des Standards möglich.

Anhang A.4 Semantic Web (RDF)

Tabelle 30: Bewertung des Semantic Web hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	5	Mit dem <i>SubClass</i> - bzw. <i>SubProperty</i> -Ansatz von <i>RDF-Schema</i> ist es möglich, hierarchische Ontologien zu definieren. <i>Statements</i> stellen Zusammenhänge zwischen Objekten in einer semantisch aussagefähigen Form dar.
1b	Modularität	5	Informationen können in relativ beliebiger Form in verschiedenen, flexibel miteinander kombinierbaren Dokumenten dargestellt werden.
1c	Verteilte Darstellung und Speicherung	1	Da kein Code integriert werden kann, können abgekoppelte Betriebszustände mit reinem RDF als Kontextmodell nicht angemessen unterstützt werden.
1d	Unabhängigkeit von Programmiersprachen	5	Die Verwendung von RDF ist nicht an spezifische Programmiersprachen gebunden.
1e	Flexible Adressierungsmöglichkeiten	4	RDF ist in der Lage, beliebige Assoziationen zwischen Informationen darzustellen, die allerdings lediglich statischer Natur sind.
2a	Separate Strukturierung von Datendimensionen	5	Separate Datendimensionen können per eigener Ontologie definiert werden.
2b	Kombination separat strukturierter Datendimensionen	5	Multiple Vererbung ist möglich.
3a	Dynamische Elemente	1	Inhalte in RDF sind rein statischer Natur, so dass keine Mechanismen zur Definition dynamischer Eigenschaften vom Standard bereitgestellt werden.
3b	Generalisierende Ontologien	5	Die in RDF verwendeten Metadatenstrukturen sind flexibel hinsichtlich aller geforderten Kriterien veränderbar.
3c	Darstellung der Historie von Kontextvariablen	1	Eine Darstellung der Historie von Properties ist nicht vorgesehen.
4	Datenqualität	2	Erweiterungen zur Beschreibung der Datenqualität sind in RDF nicht vorgesehen, könnten jedoch mit den Mitteln von RDF nachgebildet werden.
5	Schutz der Privatsphäre	2	RDF sieht keine Zugriffsmodell auf Daten vor, welches jedoch sehr einfach realisiert werden könnte.
6a	Realweltobjektbezogene Struktur	4	Realweltobjekte werden in RDF durch <i>resources</i> dargestellt, zu denen mittels <i>properties</i> Kontextvariablen hinsichtlich frei definierbarer Aspekte zugeordnet werden können. Allerdings wird die Zuordnung von Attributen zu Realweltobjekten bei RDF ausgehend von den jeweiligen Kontextvariablen vorgenommen, nicht sichergestellt werden kann, ausgehend von Realweltobjektmodellen sämtliche damit assoziierten Kontextvariablen zugreifen zu können.
6b	Eineindeutige Realweltobjektmodelle	5	Realweltobjekte werden aus dem Kontextmodell heraus über eine URI identifiziert, wobei modellseitig davon ausgegangen wird, dass jedem Realweltobjekt nur eine URI zugeordnet wird.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6c	Explizite Ontologie	5	Mit RDF-Schema steht ein expliziter ontologischer Mechanismus zur Verfügung.
6d	Darstellung von Kontextquellen	2	Kontextquellen können durch Ressourcen dargestellt werden, aus diesen Quellen gewonnene Aussagen können diesen zugeordnet werden. Über <i>reified statements</i> ist darüber hinaus die Zuordnung verschiedener, z. T. widersprüchlicher subjektiver Aussagen zu einzelnen Kontextquellen möglich. Allerdings sind diese Verfahren nicht explizit standardisiert.
6e	Inhaltliche Darstellung von Kontextvariablen	3	Externe Inhalte können innerhalb eines RDF-Modells dargestellt werden, jedoch ist die Darstellung von Mechanismen zur Automatisierung dieses Vorhabens nicht möglich.
6f	Verbindung vom Kontextmodell zu den Originaldaten	2	RDF unterstützt die Zuordnung von Properties zu Datensätzen in externen Kontextquellen nicht im speziellen.
6g	Fusion von Kontextvariablen	1	RDF unterstützt nur die Darstellung statischer Inhalte und damit keine Operationen auf Inhalten, etwa um deren Inhalte zu vereinigen.
6h	Integration von Methodik in das Datenmodell	1	RDF ist ein reines Datenformat und unterstützt keinerlei methodische Elemente.
6i	Darstellung des Datenpfades	1	Da keine Transformationen auf Inhalten dargestellt werden können, kann auch kein Datenpfad abgebildet werden.
6j	Einschränkung der Sichtbarkeit von Konzepten	3	Bereiche der Datenbasis können nur über <i>XML-Namespaces</i> abgegrenzt werden. Feingranularere oder kontextbezogene Abgrenzungen sind jedoch nicht möglich.
6k	Synonyme Benennung von Topics	2	Prinzipiell können mittels <i>sequences</i> mehrere Namen für Ressourcen angegeben werden, jedoch ist dies nicht genau genug festgelegt, so dass nicht von einer standardisierten Lösung gesprochen werden kann.
7a	Darstellung verschiedener Abstraktionsstufen	2	Verschiedene Abstraktionsstufen als Konzept existieren in RDF nicht. Ein Nachbau wäre möglich, aber proprietär.
7b	Endnutzertaugliche Programmiermethodiken	3	RDF-Modelle können in einzelne, relativ leicht verständliche Aussagen untergliedert werden. Allerdings wird dieser Ansatz bei einer steigender Anzahl von Statements schnell unübersichtlich. Zudem fehlen wichtige Elemente, um den Datenpfad strukturiert und übersichtlich darstellen zu können. Auch gelten diese Aussagen nur für den Datenteil, da methodische Elemente komplett fehlen.
7c	Problemspezifisch Sichtweisen	2	Subjektive Sichtweisen auf die Datenbasis sind mit den durch RDF bereitgestellten Werkzeugen nicht realisierbar, sondern nur durch proprietäre Erweiterungen.

Anhang A.5 Topic Maps

Tabelle 31: Bewertung des *Topic Map*-Ansatzes hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	5	Topic Maps (TM) erlauben die hierarchische Definition sowohl von Topics als auch von Assoziationen.
1b	Modularität	5	Der TM-Standard sieht die Kombination mehrerer Maps sowie die Adressierung von Informationen in anderen Maps vor.
1c	Verteilte Darstellung und Speicherung	1	Für funktionierende Repliken müssten TM hinsichtlich der Darstellung von Inhalten und Code erweitert werden.
1d	Unabhängigkeit von Programmiersprachen	5	TM stellen reine Datenstrukturen dar, die unabhängig von der konkreten Programmiersprache verwendet werden können.
1e	Flexible Adressierungsmöglichkeiten	4	TM erlauben die flexible Darstellung beliebiger Assoziationen zwischen beliebigen Topics, die jedoch lediglich einen statischen Charakter haben.
2a	Separate Strukturierung von Datendimensionen	4	TM unterstützen die Definition separater hierarchischer Ontologien.
2b	Kombination separat strukturierter Datendimensionen	4	Multiple Vererbung erlaubt die flexible Kombination heterogener Ontologien.
3a	Dynamische Elemente	1	Der TM-Standard sieht nicht einmal die Darstellung der Inhalte von Variablen selbst und damit auch keinerlei Aussagen über deren Veränderlichkeit vor.
3b	Generalisierende Ontologien	5	Die in TM verwendeten Metadatenstrukturen sind flexibel und in vielen Richtungen erweiterbar.
3c	Darstellung der Historie von Kontextvariablen	1	Der TM-Standard sieht keine Speicherung von Inhalten vor, womit auch deren Historisierung nicht betrachtet wird.
4a	Datenqualitätsdeskriptoren	1	Der TM-Standard sieht keine Unterscheidung qualitativ unterschiedlicher Attributwerte vor. Da Attributwerte nur über typisierbare, aber nicht weiter beschreibbare Occurrences dargestellt werden können, ist eine Darstellung von Datenqualitätsattributen innerhalb von TM schwierig.
4b	Datentypspez. Qualitätsdeskriptoren	1	Da keine Datenqualitätsattribute dargestellt werden können, können sie auch nicht hinsichtlich verschiedener Kontextvariablenarten unterschieden werden.
5a	Datenschutzdeskriptoren	2	Datenschutzattribute können nur auf der Ebene von Realweltobjekten, nicht jedoch für einzelne Attribute definiert werden. Dies ist jedoch nicht standardisiert.
5b	Darstellung von Akteuren	2	Akteure könnten durch Topics dargestellt werden, was jedoch nicht standardisiert ist.
5c	Vertrauensstufen	2	Vertrauensstufen könnten durch Topics dargestellt werden, was jedoch nicht standardisiert ist.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6a	Realweltobjekt-bezogene Struktur	5	TM organisieren die Informationen ausgehend von Realweltobjektrepräsentationen. Ausgehend von diesen können sämtliche dazugehörigen Kontextinformationen zugegriffen werden.
6b	Eineindeutige Realweltobjektmodelle	5	Über <i>Published Subject Indicators</i> können globale Identifikatoren für Topics bekannt gemacht werden. Darüber lässt sich bei Kenntnis eines eindeutigen Identifikators für Realweltobjekte auch das passende Topic für ein Realweltobjekt finden. Bei der Zusammenführung von TM werden synonyme Topics zusammengefasst.
6c	Explizite Ontologie	5	TM stellen selbst eine Methode zur Darstellung ontologischer Informationen über externe Informationsmengen dar. Dabei können sie selbst wiederum auf expliziten, ebenfalls durch Topic Maps dargestellten Ontologien basieren.
6d	Darstellung von Kontextquellen	2	Topic Maps wurden speziell für die Beschreibung heterogener Datenstrukturen aus verschiedenen Quellen entworfen, adressieren diese jedoch direkt und nicht über explizite Darstellungen der jeweiligen Datenquelle. Die Angabe von automatisch zum Zugriff auf diese Quellen zu verwendenden Informationen ist auch nicht möglich.
6e	Inhaltliche Darstellung von Kontextvariablen	1	Externe Inhalte können von Topic Maps nur referenziert, nicht jedoch inhaltlich dargestellt werden.
6f	Verbindung vom Kontextmodell zu den Originaldaten	4	Topic Maps können externe Ressourcen über <i>Occurrences</i> , die eine URI enthalten, referenzieren. Logindaten aus einem expliziten Modell der betreffenden Datenquelle können jedoch nicht zum automatisierten Zugriff auf die externe Datenquelle verwendet werden.
6g	Fusion von Kontextvariablen	1	Da keine Inhalte in TM dargestellt werden, erübrigt sich auch die Kombination von deren Inhalten.
6h	Integration von Methodik in das Datenmodell	1	TM unterstützen keine methodischen Elemente etwa zur Transformation von Inhalten.
6i	Darstellung des Datenpfades	1	Da keine Transformationen dargestellt werden können, erübrigt sich auch die Darstellung eines Datenpfades.
6j	Einschränkung der Sichtbarkeit von Konzepten	5	Die Sichtbarkeit von Topics, Assoziationen bzw. beliebigen anderen Komponenten kann durch Angabe eines <i>Scope</i> sehr flexibel und unabhängig von Modulgrenzen eingeschränkt werden.
6k	Synonyme Benennung von Topics	5	Für jeden Topicnamen können verschiedene <i>Varianten</i> angegeben werden, die u. U. nur in bestimmten Zusammenhängen sichtbar sind.
7a	Darstellung verschiedener Abstraktionsstufen	2	Das Konzept von Abstraktionsstufen ist nicht im TM-Standard enthalten, kann jedoch recht einfach realisiert werden.
7b	Endnutzertauglich Programmiermethodiken	4	TM stellen die enthaltenen Informationen auf einem recht hohen Niveau mittels untereinander vernetzter Realweltobjektmodelle dar. Dies gilt allerdings nur für den Datenteil und nicht die Methodik zur Transformation der Daten.
7c	Problemspezifische Sichtweisen	5	Topic Maps können in mehreren Ebenen angeordnet und so subjektive Sichtweisen auf die Datenbasis erstellt werden. Die so realisierten Modelle wären in jeder TM-Engine lauffähig, so dass an dieser Stelle keine Erweiterungen des Standards notwendig wären.

Anhang A.6 Context Cubes

Tabelle 32: Bewertung des *Context Cube*-Ansatzes hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	5	Context Cubes (CC) stellen ein ausreichend formalisiertes, kontextbezogenes Datenmodell zur Verfügung. Generalisierungsbeziehungen werden über <i>context dimension hierarchies</i> dargestellt.
1b	Modularität	1	Die in CC enthaltenen Informationen können nicht in kleinere Einheiten aufgeteilt oder diese zu größeren Modellen kombiniert werden.
1c	Verteilte Darstellung und Speicherung	1	Data Cube-Modelle sind aufgrund der fehlenden Integration von Methodik nicht funktional vollständig replizierbar.
1d	Unabhängigkeit von Programmiersprachen	5	Aufgrund des datenbezogenen Charakters stellen CC keine besonderen Anforderungen an zu verwendende Programmiersprachen.
1e	Flexible Adressierungsmöglichkeiten	3	Assoziationen werden durch Tupel von Fremdschlüsseln dargestellt, weshalb nur die vom Schema definierten Assoziationen möglich sind.
2a	Separate Strukturierung von Datendimensionen	5	Das bei <i>Context Cubes</i> verwendete Data Cube-Datenmodell ist prinzipbedingt multidimensional. Es unterstützt separate Ontologien für jede einzelne Datendimension, welche in Form von <i>context dimension hierarchies</i> erfolgen.
2b	Kombination separat strukturierter Datendimensionen	5	Das CCDM erlaubt die flexible Kombination separater Dimensionen zu mehrdimensionalen Datenräumen.
3a	Dynamische Elemente	1	Das CC-Datenmodell kann keine dynamischen Elemente darstellen.
3b	Generalisierende Ontologien	5	Die verwendeten Datenmodelle können zu späterer Zeit erweitert werden, wobei bei einer entsprechenden <i>roll-up</i> -Operation auch die neuen Informationen durch existierende Anwendungen verwendet werden können.
3c	Darstellung der Historie von Kontextvariablen	5	CC sehen Historie als eine immer zu verwendende Kontextdimension an.
4a	Datenqualitätsdeskriptoren	2	Das CC-Datenmodell sieht keine Datenqualitätsattribute und diesbezüglichen Auswertungsmechanismen vor, zumindest erstere können jedoch in Form von zusätzlichen Kontextdimensionen hinzugefügt werden.
4b	Datentypspez. Qualitätsdeskriptoren	2	Da typisierte Datenattribute unterschieden werden, könnten ihnen Datenqualitätsattribute in individueller Form hinzugefügt werden.
5	Schutz der Privatsphäre	2	Datenschutzmechanismen sind im CC-Datenmodell nicht vorgesehen, könnten jedoch auf proprietäre Weise hinzugefügt werden.
6a	Realweltobjekt-bezogene Struktur	5	Realweltobjekte werden im beim CCDM verwendeten Star-Schema durch jeweils eigene Tabellen (<i>context attribute tables</i>) dargestellt, die sämtliche Attribute zu diesem Realweltobjekt darstellen.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6b	Eineindeutige Realweltobjektmodelle	3	Die Realweltobjekte sollen modellseitig eindeutig ihrer jeweiligen Repräsentation im CC-Modell zugeordnet werden, allerdings werden vom CCDM dafür keine speziellen Mechanismen bereitgestellt.
6c	Explizite Ontologie	5	Realweltobjekte werden mittels <i>context dimension hierarchies</i> , Assoziationen mittels <i>Star-Schemata</i> modelliert. Allerdings können.
6d	Darstellung von Kontextquellen	5	Die beim CCDM verwendeten Data Warehouse-Technologien beinhalten den Zugriff auf externe Datenquellen sowie die Transformation der erfassten Daten in das benötigte Format.
6e	Inhaltliche Darstellung von Kontextvariablen	5	CC kopieren sämtliche erfassten Kontextwerte in eine interne Datenbasis und stellen Methoden bereit, um die interne Repräsentation aktuell zu halten.
6f	Verbindung vom Kontextmodell zu den Originaldaten	4	Links auf externe Daten werden im Data Warehouse-Datenmodell mittels <i>Monitor-</i> und <i>Extractor-</i> Komponenten dargestellt, die allerdings nicht Bestandteil des eigentlichen Kontextmodells sind.
6g	Fusion von Kontextvariablen	4	Die von verschiedenen Extraktoren gelieferten Informationen werden durch eine <i>Integrator</i> genannte Komponente in das DWH eingepflegt, wobei sie zuerst auch verarbeitet werden können. Allerdings sind diese Informationen nicht Bestandteil des eigentlichen Kontextmodells.
6h	Integration von Methodik in das Datenmodell	3	Logik zur Transformation von Datenbankinhalten kann im CCDM lediglich in Form von Integrator-Code dargestellt werden. Dieser wird jedoch nicht im Datenmodell dargestellt. Eine Integration unterschiedlicher Kontextquellen kann daher nur zum Beschaffungszeitraum erfolgen, weshalb im Modell auch nur die Ergebnisse, jedoch nicht die jeweiligen Originaldaten verfügbar sind.
6i	Darstellung des Datenpfades	1	Da innerhalb des Datenmodells keine Operationen dargestellt werden können, kann auch kein Datenpfad abgebildet werden.
6j	Einschränkung der Sichtbarkeit von Konzepten	1	Daten können nicht auf bestimmte Bereiche des CCDM eingegrenzt werden. Es ist lediglich möglich, bestimmte Detailinformationen in höher verdichteten Sichtweisen auszublenden, jedoch wird damit nicht diese Anforderung erfüllt.
6k	Synonyme Benennung von Topics	2	Synonyme Objektamen sind im CCDM nicht vorgesehen, können aber durch entsprechende Attribute proprietär dargestellt werden.
7a	Darstellung verschiedener Abstraktionsstufen	3	Das CC-Datenmodell stellt außer einer inhaltlichen Verdichtung entlang vordefinierter Granularitätsstufen (<i>roll-up</i> bzw. <i>drill-down</i>) keine wirklich unabhängigen Abstraktionsstufen von Daten bereit.
7b	Endnutzertauglichkeit	3	Die beim CCDM verwendeten multidimensionalen Datenwürfel sowie die darauf anzuwendenden Operationen sind eine ohne grundlegende Einführung in die <i>Data Cube</i> -Technologie nicht verwendbar. Ansonsten werden relativ intuitive Konzepte verwendet.
7c	Problemspezifische Sichtweisen	1	Das CCDM stellt eine einzelne globale Sichtweise auf die Datenbasis dar. Die Sichtweise kann lediglich durch <i>slice-</i> bzw. <i>dice</i> -Operationen eingengt werden, wodurch jedoch nicht die in dieser Anforderung geforderte ontologische Subjektivierung der Daten realisiert werden kann.

Anhang A.7 Henricksen

Tabelle 33: Bewertung des Ansatzes von HENRICKSEN hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	2	Über kontextbezogene Ontologien werden von <i>Henricksen</i> keine Aussagen gemacht, so dass nicht davon ausgegangen werden kann, dass solche existieren. Formalisiert sind lediglich einige eher modellierungsbezogene Assoziationsarten etwa zur Darstellung der Kardinalität, jedoch keine kontextbezogenen Aspekte.
1b	Modularität	2	Eine Modularisierung des Kontextmodells wird in den Unterlagen zum Modell nicht erwähnt. Eine Kombination separater Teilmodelle ist modellseitig nicht vorgesehen.
1c	Verteilte Darstellung und Speicherung	2	In der Datenbasis kann angegeben werden, welche Transformationen durch welche (externen) Code-segmente ausgeführt worden sind. Eine Einbettung von Code in die Datenstruktur ist jedoch nicht vorgesehen. Daher ist das Modell nicht funktionsfähig replizierbar.
1d	Unabhängigkeit von Programmiersprachen	5	Aufgrund seines datenbezogenen Charakters ist das Kontextmodell unabhängig von den zum Zugriff auf das Modell verwendeten Programmiersprachen.
1e	Flexible Adressierungsmöglichkeiten	4	Im Modell können beliebige Assoziationen dargestellt werden, die zwar benannt, nicht jedoch in einer expliziten Ontologie definiert werden können und lediglich einen statischen Charakter haben.
2a	Separate Strukturierung von Datendimensionen	2	Das Modell basiert auf lediglich einer impliziten Ontologie. Multidimensionalität wird modellseitig nicht unterstützt.
2b	Kombination separater strukturierter Datendimensionen	2	Eine Kombination verschiedener Ontologien in multidimensionale Kontextmodelle ist aufgrund der fehlenden Unterstützung für Multidimensionalität nicht möglich, kann jedoch mittels der ausdrucksstarken Assoziativität des Modells proprietär realisiert werden.
3a	Dynamische Elemente	4	Die Kennzeichnung von durch Sensoren erfassten sowie über Aktualisierungsabhängigkeiten daraus abgeleiteten Kontextvariablen ermöglicht die dynamische Verwendung veränderlicher Inhalte. Eine weitergehende Unterstützung von Dynamik, etwa strukturelle Veränderungen der Datenbasis zur Laufzeit, sind jedoch nicht vorgesehen.
3b	Generalisierende Ontologien	1	Da keine Ontologien existieren, ist die Erweiterung der ontologischen Grundlage der Datenbasis zur Laufzeit nicht direkt möglich.
3c	Darstellung der Historie von Kontextvariablen	3	Zwar können über <i>temporal associations</i> zeitlich verschiedene Werte für eine Kontextvariable angegeben werden, es ist jedoch fraglich, ob dies für eine brauchbare Angabe historischer Daten taugt bzw. überhaupt so vorgesehen ist.
4	Datenqualität	5	Henricksen verwendet mit dem Qualitätsmodell von WANG ein sehr flexibles Datenqualitätsmodell, das sämtliche Anforderungskriterien erfüllt (vgl. [Hen ⁺ 02, S. 178], [Wan ⁺ 95]).
5	Schutz der Privatsphäre	2	Die Modellierung von Zugriffsmodalitäten auf einzelne Kontextinformationen wird nicht modelliert, könnte jedoch analog zu der Modellierung der Datenqualität implementiert werden.
6a	Realweltobjektbezogene Struktur	5	Realweltobjekte werden durch <i>Entity</i> genannte Objektstrukturen dargestellt, denen über spezielle Assoziationsarten Kontextvariablen in Form von Attributen zugeordnet werden.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6b	Eineindeutige Realweltobjektmodelle	4	Jedes Objekt im Datenmodell soll genau ein Realweltobjekt beschreiben und umgekehrt. Allerdings ist diese Zuordnung nicht formalisiert, da konkrete Zeiger auf Realweltobjekte bzw. umgekehrt fehlen.
6c	Explizite Ontologie	2	Das Kontextmodell basiert auf einer Ontologie, jedoch wird diese nicht explizit dargestellt, sondern implizit von den Clientanwendungen verwendet.
6d	Darstellung von Kontextquellen	2	Externe Kontextquellen sind im Kontextmodell nicht vorgesehen, könnten aber als Entities, welche durch spezielle Assoziationen mit ihren entsprechenden Kontextvariablen verbunden sind, dargestellt werden.
6e	Inhaltliche Darstellung von Kontextvariablen	4	Das Kontextmodell kann beliebige Kontextinformationen inhaltlich darstellen. Der einzige Mechanismus zur Aktualisierung dieser Inhalte stellt deren Kennzeichnung als von externen Quellen erfasst dar.
6f	Verbindung vom Kontextmodell zu den Originaldaten	1	Aufgrund der fehlenden Darstellung von externen Datenquellen ist eine Adressierung externer Informationen nicht möglich. Externe Informationen werden im Modell ohne Informationen, wo sie herkommen, dargestellt. Zu ihnen wird lediglich vermerkt, dass sie aus einer externen Quelle stammen.
6g	Fusion von Kontextvariablen	2	Kontextvariablen können in andere Kontextvariablen transformiert werden. Dabei kann jedoch nur jeweils eine Quelle in ein Ziel transformiert werden, so dass eine inhaltliche Kombination synonymmer Kontextvariablen nicht möglich ist.
6h	Integration von Methodik in das Datenmodell	3	Modellseitig können mittels <i>derived associations</i> nur Aufrufe auf externe Codesegmente, jedoch keine eigenen Methoden dargestellt werden.
6i	Darstellung des Datenpfades	5	Die Verarbeitungskette von Kontextinformationen kann mittels <i>derived associations</i> dargestellt werden.
6j	Einschränkung der Sichtbarkeit von Konzepten	1	Eine Eingrenzung der Sichtbarkeit von Konzepten ist nicht vorgesehen und würde mehrere zusätzliche Konzepte in diesem Datenmodell erfordern.
6k	Synonyme Benennung von Topics	2	Synonyme Objektnamen sind in dem Modell nicht vorgesehen, könnten aber durch spezielle Assoziationsarten dargestellt werden.
7a	Darstellung verschiedener Abstraktionsstufen	2	Eine explizite Darstellung verschiedener logischer Abstraktionsstufen ist im Modell nicht vorgesehen, könnte jedoch durch einen zusätzlichen Assoziationstyp realisiert werden.
7b	Endnutzertaugliche Programmiermethodiken	1	Das Kontextmodell stellt eine ganze Reihe von <i>low level</i> -Modellinformationen, jedoch nicht die für die endnutzerbezogene Programmierung notwendigen höherwertigen Informationen dar. Zudem ist das Modell aufgrund der vielen, teilweise miteinander zu kombinierenden Assoziationsarten sehr unübersichtlich und kann die Komplexität nicht angemessen reduzieren.
7c	Problemspezifisch Sichtweisen	1	Das Kontextmodell kann aufgrund seines globalen, nicht auf expliziten Ontologien basierenden Charakters keine anwendungsspezifischen Sichtweisen darstellen.

Anhang A.8 Context Toolkit

Tabelle 34: Bewertung des Datenmodells des *Context Toolkit* hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	4	Im objektorientierten Kontextmodell des <i>Context Toolkit</i> stellen <i>Widgets</i> Kontextvariablen dar, die mittels <i>Aggregatoren</i> zu Realweltobjektrepräsentationen zusammengefasst werden. Es können keine Assoziationen dargestellt werden.
1b	Modularität	3	Das Kontextmodell einer Anwendung kann aus verschiedenen <i>Widgets</i> und <i>Aggregatoren</i> zusammengesetzt werden. Andere Unterteilungen des Modells sind jedoch nicht möglich.
1c	Verteilte Darstellung und Speicherung	5	Im Kontextmodell des <i>Context Toolkit</i> sind sämtliche Elemente enthalten, um funktionsfähige Repliken von Teilen davon auf anderen Endgeräte zu erstellen.
1d	Unabhängigkeit von Programmiersprachen	1	Zwar werden intern mit <i>Web Services</i> auch programmiersprachenunabhängige Konzepte angedacht, die Interaktion mit dem <i>Context Toolkit</i> erfolgt wegen des objektorientierten Ansatzes i. d. R. jedoch mittels einer Objektklasse, die von Client-Anwendungen instanziiert werden muß, weshalb auch diese in Java geschrieben sein müssen (vgl. [Dey ⁺ 01b, S. 28 ff]).
1e	Flexible Adressierungsmöglichkeiten	2	Kontextvariablen können lediglich mittels <i>Aggregatoren</i> zu Realweltobjektrepräsentationen zusammengefasst werden, jedoch können keine Verbindungen zwischen verschiedenen <i>Aggregatoren</i> dargestellt werden. Damit sind keine indirekten Adressierungsmöglichkeiten gegeben.
2a	Separate Strukturierung von Datendimensionen	5	Verschiedene Kontextdimensionen können im Kontextmodell des <i>Context Toolkit</i> mittels verschiedener <i>Widgets</i> nebeneinander dargestellt werden. Ein Beispiel findet sich in [Dey00b, S. 49].
2b	Kombination separat strukturierter Datendimensionen	5	Die in verschiedenen <i>Widgets</i> definierten Ontologien können mittels <i>Aggregatoren</i> miteinander kombiniert werden.
3a	Dynamische Elemente	3	Im Prinzip werden sämtliche Inhalte als dynamisch angenommen. Allerdings fehlen detaillierte Mechanismen zur Beschreibung der durchaus unterschiedlichen Dynamik verschiedener Datenfelder.
3b	Generalisierende Ontologien	1	Die verwendeten Ontologien sind nichthierarchisch und damit nicht in dem an dieser Stelle geforderten Maß erweiterbar.
3c	Darstellung der Historie von Kontextvariablen	5	Aufgezeichnete Informationen werden von den <i>Widgets</i> gespeichert und können bei Bedarf abgerufen werden.
4	Datenqualität	2	Die Behandlung mehrdeutiger Messwerte soll durch einen vom <i>Nutzer</i> durchgeführten Vermittlungsprozess aufgelöst werden (vgl. [Dey00b, S. 94 ff]). Da explizite Datenqualitätsattribute von <i>Widgets</i> fehlen, können aus fehlerhaften Daten resultierende Widersprüche nur erkannt, aber nicht automatisiert aufgelöst werden. Dieser Ansatz wird bei den zu erwartenden umfangreichen Kontextinformationsmengen jedoch als nicht ausreichend angesehen (vgl. [Hen ⁺ 02, S. 178]).
5	Schutz der Privatsphäre	2	Datenschutzmechanismen werden nicht erwähnt, könnten im Prinzip jedoch als zusätzliche <i>Widgets</i> bzw. Attribute davon realisiert werden.
6a	Realweltobjektbezogene Struktur	3	Bei entsprechender Modellierung enthalten <i>Aggregatoren</i> sämtliche zu einem Realweltobjekt gehörenden <i>Widgets</i> . Dies ist jedoch nicht zwingend vorgeschrieben.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6b	Eineindeutige Realweltobjektmodelle	2	Das <i>Context Toolkit</i> ist zwar in der Lage, elementare Kontextvariablen zu beliebigen Informationsobjekten zusammenzufassen, modellseitig werden jedoch keine Mechanismen vorgesehen, um Realweltobjekte oder -modelle global eindeutig zu identifizieren. Aggregatoren stellen vielmehr ein optionales Instrument dar, um die voneinander völlig unabhängigen Widgets strukturell zumindest grob zu organisieren.
6c	Explizite Ontologie	1	Widgets selbst stellen beliebige Messwerte dar. Deren ontologische Interpretation obliegt den jeweiligen Client-Anwendungen und wird daher in der Methodologie des <i>Context Toolkit</i> weder explizit dargestellt noch ist sie damit zwischen verschiedenen Anwendungen austauschbar.
6d	Darstellung von Kontextquellen	2	Externe Daten werden über <i>Widgets</i> besorgt. Diese werden individuell konfiguriert. Eine Konfiguration über separate Kontextquellenmodelle, welche Zugangsdaten in wiederverwendbarer Form enthalten, ist nicht vorgesehen.
6e	Inhaltliche Darstellung von Kontextvariablen	5	Sensorwerte werden zeitnah in das interne Kontextmodell kopiert und können dort weiterverarbeitet werden.
6f	Verbindung vom Kontextmodell zu den Originaldaten	4	Kontextquellen werden über <i>Widgets</i> repräsentiert und zugegriffen. Sämtliche Adressierungsinformationen werden innerhalb der <i>Widgets</i> dargestellt.
6g	Fusion von Kontextvariablen	4	Inhalte von Kontextvariablen können in <i>Interpretern</i> beliebig miteinander kombiniert werden.
6h	Integration von Methodik in das Datenmodell	4	Operationen auf Inhalten im Kontextmodell werden durch <i>Interpreter</i> dargestellt. Diese sind allerdings extern und nicht an der entsprechenden Stelle des Datenmodells dargestellt (vgl. 4.3.1 („Externe Agenten“, S. 108)).
6i	Darstellung des Datenpfades	5	Der Datenpfad kann aus den Informationen im Kontextmodell rekonstruiert werden.
6j	Einschränkung der Sichtbarkeit von Konzepten	1	Es ist weder möglich, logische Bereiche in der Datenbasis zu definieren, noch die Sichtbarkeit von Aggregatoren oder <i>Widgets</i> in irgend einer Form einzugrenzen.
6k	Synonyme Benennung von Topics	2	Synonyme Namen für Aggregatoren sind nicht vorgesehen.
7a	Darstellung verschiedener Abstraktionsstufen	3	Durch Interpretation können unterschiedlich abstrakte Kontextinformationen dargestellt werden. Modellseitig werden jedoch keine separaten, voneinander getrennten Abstraktionsebenen für die Inhalte des Kontextmodells unterstützt.
7b	Endnutzertaugliche Programmiermethodiken	1	Endnutzern könnte der Zugriff auf den innerhalb von <i>Interpretern</i> verwendeten Code oder die Struktur von Aggregatoren erlaubt werden.
7c	Problemspezifische Sichtweisen	3	Subjektive Sichtweisen können mittels zusätzlicher <i>Widgets</i> und Aggregatoren definiert werden, die Informationen aus dem Kontextmodell beschaffen und transformieren. Dieser Ansatz wird vom Metamodellell jedoch nicht weitergehend vorgesehen.

Anhang A.9 Context Fabric

Tabelle 35: Bewertung des *Context Fabric* hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	5	Die Modellelemente sind aufgrund des hohen Automatisierungsgrades stark formalisiert.
1b	Modularität	1	Das Prinzip von <i>Context Fabric</i> sieht vor, alle Kontextvariablen in einem Pool vorrätig zu halten und automatisch die für eine Anfrage geeignetesten Kontextvariablen zu verwenden. Daher sind keine modularen Konzepte vorhanden.
1c	Verteilte Darstellung und Speicherung	4	Eine Replikation von Teilen des Kontextmodells auf verschiedene Endgeräte ist möglich, vorausgesetzt die entsprechenden Transformatoren werden ebenfalls kopiert.
1d	Unabhängigkeit von Programmiersprachen	5	Analog zum Ansatz von SQL können Abfragen aus jeder Programmiersprache heraus in einer deskriptiven Form gestellt werden, wobei die Anfragen bei <i>Context Fabric</i> in XML formuliert sind.
1e	Flexible Adressierungsmöglichkeiten	5	Realweltobjektmodelle können im <i>Context Data Model</i> flexibel miteinander verbunden werden.
2a	Separate Strukturierung von Datendimensionen	5	Es werden verschiedene Kontextdimensionen unterstützt. Für jede Kontextdimension, etwa den Ort, können auf verschiedenen Ontologien basierende Darstellungsformen verwendet werden, die auch ineinander konvertiert werden können.
2b	Kombination separat strukturierter Datendimensionen	3	Dimensionen werden vor allem separat abgefragt. Eine weitergehende Kombination verschiedener Kontextdimensionen ist nicht vorgesehen.
3a	Dynamische Elemente	3	Sämtliche Modellelemente werden als dynamisch angenommen, diesbezüglich allerdings nicht weitergehend unterschieden oder beschrieben.
3b	Generalisierende Ontologien	4	Ontologien können dank ihrer starken Formalisierung zu späterer Zeit auch für existierende Anwendungen verständlich erweitert werden.
3c	Darstellung der Historie von Kontextvariablen	5	Vergangene Werte von Kontextvariablen werden im Modell zusammen mit einem Zeitstempel dargestellt (vgl. [Hon ⁺ 03, S. 13]).
4a	Datenqualitätsdeskriptoren	5	Für Kontextdaten können Datenqualitätsattribute, z. B. deren Wahrscheinlichkeit angegeben werden (vgl. [Hong01, S. 7]).
4b	Datentypspez. Qualitätsdeskriptoren	2	Außer einer prozentualen Angabe für die Wahrscheinlichkeit wird nicht auf andere Darstellungsformen von Datenqualität oder deren datentypspezifische Gestaltung eingegangen.
5a	Datenschutzdeskriptoren	5	Zugriffsschutz wird über entsprechende Attribute realisiert (vgl. [Hon ⁺ 03, S. 13]).
5b	Darstellung von Akteuren	5	In [Hon ⁺ 03, S. 57] werden verschiedene Akteure mit unterschiedlichen Vertrauensstufen erwähnt.
5c	Vertrauensstufen	5	Es werden u. a. verschiedene Detaillierungsstufen für unterschiedlich vertrauenswürdige Situationen vorgesehen.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6a	Realweltobjekt-bezogene Struktur	5	Das Kontextmodell beinhaltet Realweltobjektmodelle und diesen zugeordnete Attribute (vgl. [Hon ⁺ 03, S. 73]).
6b	Eineindeutige Realweltobjektmodelle	2	Über eineindeutige Zuordnungen von Realweltobjekten zu ihren Modellen in <i>Context Fabric</i> wird nichts ausgesagt. Zur Identifikation werden keine eineindeutigen Identifikatoren verwendet, etwa die EMail-Adresse zur Identifikation von Personen, wobei jede Person jedoch mehrere solcher Adressen haben kann.
6c	Explizite Ontologie	5	Es werden <i>Schema</i> genannte Ontologien verwendet, die auch explizit dargestellt werden. Kontextvariablen können in verschiedene Schemata konvertiert werden.
6d	Darstellung von Kontextquellen	4	Es werden verschiedene Datenquellen unterschieden (vgl. [Hon ⁺ 03, S. 7, S. 72]), für die wie in [Hon ⁺ 03, S. 111] dargestellt auch Zugriffsdaten modelliert werden können. Dies gilt jedoch nur für XML-strukturierte, d. h. per Web Service zugreifbare Informationen.
6e	Inhaltliche Darstellung von Kontextvariablen	5	Inhalte von externen Kontextquellen werden zeitnah ins Kontextmodell übertragen.
6f	Verbindung vom Kontextmodell zu den Originaldaten	4	Originaldaten in externen Datenquellen können lediglich per URL adressiert werden. Andere Adressformate werden nicht unterstützt.
6g	Fusion von Kontextvariablen	3	Kontextvariablen mit gleichem Inhalt können im Modell dargestellt werden, wobei zur Laufzeit eine Auswahl der geeignetesten Variablen erfolgt. Eine inhaltliche Integration synonyme Variablen findet jedoch nicht statt.
6h	Integration von Methodik in das Datenmodell	3	Im Kontextmodell selbst können nur Transformationen, aber keine anderen Formen von Methodik dargestellt werden.
6i	Darstellung des Datenpfades	5	Der Datenpfad wird nicht nur flexibel zur Laufzeit bestimmt, sondern kann auch durch die Kombination der verwendeten <i>Transformatoren</i> dargestellt werden.
6j	Einschränkung der Sichtbarkeit von Konzepten	1	Das Prinzip von <i>Context Fabric</i> sieht einen globalen Pool von Sensoren und Transformatoren vor, wobei der jeweils zu verwendende Sensor zur Laufzeit bestimmt wird. Einschränkungen auf explizit definierte semantische Ausschnitte sind im Modell nicht vorgesehen.
6k	Synonyme Benennung von Topics	5	Objekte können in Anfragen über verschiedene Namen adressiert werden.
7a	Darstellung verschiedener Abstraktionsstufen	1	Modellseitig werden keine separaten Abstraktionsebenen für ein und die selbe Informationsmenge vorgesehen.
7b	Endnutzertauglich Programmiermethodiken	4	Da die internen Abläufe vollautomatisiert ablaufen, können Endnutzer dort nicht eingreifen. Die Änderungen wären auch schwierig zu realisieren. Es ist aber möglich, neue Datenquellen oder Transformatoren hinzuzufügen.
7c	Problemspezifische Sichtweisen	4	Die Nutzung verschiedener Ontologien auf der Ebene der Abfrageschnittstelle ist dank der flexiblen Transformationstechnik möglich, wird allerdings nicht in expliziter Form vom Kontextmodell vorgesehen.

Anhang A.10 Sentient Computing

Tabelle 36: Bewertung von *Sentient Computing* hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	5	Kontext wird durch ein um multiple Vererbung erweitertes ERM dargestellt, wodurch alle Anforderungen erfüllt werden.
1b	Modularität	4	Durch den objektorientierten Ansatz können Teilmodelle in einzelnen <i>Sentient Objects</i> gekapselt werden, allerdings ist mit diesem Ansatz keine freie Kombinierbarkeit der Teilmodelle möglich.
1c	Verteilte Darstellung und Speicherung	4	Der objektorientierte Ansatz ermöglicht eine gute Replikationsfähigkeit. Einzelne Objekte können auf verschiedene Endgeräte repliziert werden und sind dort auch ohne Netzwerkverbindung voll funktionsfähig. Dies setzt allerdings mobile Codesysteme voraus (vgl. [Spri04, S. 52 ff]).
1d	Unabhängigkeit von Programmiersprachen	1	Aufgrund des objektorientierten Ansatzes sind Anwendungen, welche die <i>Sentient Objects</i> nutzen möchten, auf die entsprechende Programmiersprache festgelegt.
1e	Flexible Adressierungsmöglichkeiten	4	Das ERM erlaubt flexible Verbindungen zwischen einzelnen Entities, die allerdings nicht dynamisch berechnet werden können.
2a	Separate Strukturierung von Datendimensionen	5	Die einzelnen Kontextmodelle werden innerhalb eines <i>Sentient Object</i> durch eine auf dem ERM basierende Modellierungstechnik, die um multiple Vererbung erweitert wurde, dargestellt. Dadurch sind heterogene Kontextontologien möglich.
2b	Kombination separat strukturierter Datendimensionen	5	Die multiplen Vererbungsmechanismen erlauben die Kombination unterschiedlicher Aspekte in konkrete Datenmodelle.
3a	Dynamische Elemente	5	<i>Sentient Objects</i> unterscheiden statischen und dynamische Informationen, wobei letztere durch verschiedene Dynamikattribute beschrieben werden und aus Performancegründen auch außerhalb der Datenbank dargestellt werden können (vgl. [Har ⁺ 01, S. 4 f]).
3b	Generalisierende Ontologien	5	Durch den hierarchischen Ansatz sind die Ontologien zur Laufzeit auch für existierende Anwendungen verständlich erweiterbar.
3c	Darstellung der Historie von Kontextvariablen	2	Eine Historisierung von Inhalten ist nicht vorgesehen, könnte jedoch durch spezielle Objekte oder auf der Datenbank-Ebene realisiert werden.
4a	Datenqualitätsdeskriptoren	3	Auf der Ebene der Sensorenverwaltung werden Bayesische Netze zur flexiblen Kombination unterschiedlich wahrscheinlicher Sensoren verwendet (vgl. [BiCa04, S. 2]). Dies schlägt sich jedoch nicht in Attributen innerhalb des Kontextmodells nieder.
4b	Datentypspez. Qualitätsdeskriptoren	2	Datenqualität spielt nur bei der Verwaltung von sensorisch erfassten Eingabedaten eine Rolle. Im Kontextmodell findet sie sich nicht wieder.
5	Schutz der Privatsphäre	2	Es werden keine Datensicherheitsmechanismen vorgesehen, diese sind aber durch entsprechende Attribute im Kontextmodell realisierbar.
6a	Realweltobjekt-bezogene Struktur	5	Durch Verwendung des ERM können Realweltobjekte als <i>Entities</i> dargestellt werden.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6b	Eineindeutige Realweltobjektmodelle	1	In jedem <i>sentient object</i> gibt es ein eigenes Kontextmodell, das durch die Eingabeinformationen des Objektes mit Inhalten gefüllt wird. Dadurch kann es mehrere Modelle von Realweltobjekten in verschiedenen <i>sentient objects</i> geben.
6c	Explizite Ontologie	4	Das Kontextmodell basiert auf einer expliziten Ontologie, allerdings können keine verschiedenen Ontologien miteinander kombiniert werden.
6d	Darstellung von Kontextquellen	2	Externe Kontextquellen können im <i>Sentient Object</i> -Modell durch separate Sentient Objects repräsentiert werden, allerdings ist dies nicht formalisiert.
6e	Inhaltliche Darstellung von Kontextvariablen	5	Inhalte aus externen Kontextquellen können in das Modell kopiert und dort beliebig verarbeitet sowie persistent gespeichert werden. Mittels entsprechender Erfassungsobjekte können die Inhalte auch automatisiert erfasst werden.
6f	Verbindung vom Kontextmodell zu den Originaldaten	1	Externe Datensätze können im Kontextmodell nicht explizit als solche dargestellt werden. Dafür wären umfangreiche Änderungen notwendig.
6g	Fusion von Kontextvariablen	4	Verschiedene Kontextvariablen können miteinander kombiniert werden, allerdings geschieht dies nicht im eigentlichen Kontextmodell, sondern bereits auf der Ebene der Sensorenverwaltung.
6h	Integration von Methodik in das Datenmodell	5	Logik kann über die Event-Condition-Action-Regeln in der kontextbasierten Inferenzmaschine eines <i>Sentient Object</i> dargestellt werden (vgl. [BiCa04, S. 3]).
6i	Darstellung des Datenpfades	3	Der Datenpfad kann begrenzt dargestellt werden, so lange er sich über verschiedene <i>Sentient Objects</i> erstreckt. Innerhalb eines solchen Objektes ablaufende Transformationen lassen sich nicht direkt darstellen.
6j	Einschränkung der Sichtbarkeit von Konzepten	3	Die Sichtbarkeit einzelner Kontextvariablen kann auf den Bereich innerhalb eines <i>Sentient Object</i> eingegrenzt werden. Weitergehende Unterteilungen der Datenbasis sind jedoch nicht möglich.
6k	Synonyme Benennung von Topics	2	Synonyme Objektamen sind in diesem Datenmodell nicht vorgesehen.
7a	Darstellung verschiedener Abstraktionsstufen	1	Verschiedene Abstraktionsstufen von Kontext sind nicht vorgesehen.
7b	Endnutzertauglich Programmiermethodiken	3	Das Modell ist nur bedingt endnutzertauglich. Das Kontextmodell ist auf viele Objekte verteilt, innerhalb der Objekte werden eher unhandliche Inferenzmaschinen verwendet.
7c	Problemspezifische Sichtweisen	1	Subjektive Sichtweisen auf die Datenbasis können im Rahmen des Modells nicht definiert werden.

Anhang A.11 Context Framework

Tabelle 37: Bewertung des Kontextmodells des *Context Frameworks* hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	4	Sämtliche Elemente des Kontextmodells sind formalisiert. Allerdings wird eine Assoziation nur über die Art der beteiligten Objekte und nicht über einen eigenen Typ identifiziert.
1b	Modularität	1	Das Kontextmodell kann nicht in einzelne, miteinander kombinierbare Teile untergliedert werden.
1c	Verteilte Darstellung und Speicherung	3	Das Kontextmodell sieht die Replikation von Informationen auf verschiedene Endgeräte vor, beschränkt dies jedoch nur auf die Daten und nicht die Methodik.
1d	Unabhängigkeit von Programmiersprachen	1	Das <i>Context Framework</i> kann nur durch <i>Java</i> -Anwendungen verwendet werden, da diese eine Klasse des Frameworks instanziiieren müssen.
1e	Flexible Adressierungsmöglichkeiten	4	Über die Möglichkeit, verschiedene Kontexte eines Objektes zu beschreiben steht eine flexible Möglichkeit zur Adressierung von Realweltobjektmodellen zur Verfügung. Allerdings können diese Assoziationen nicht dynamisch zur Laufzeit berechnet werden.
2a	Separate Strukturierung von Datendimensionen	5	Verschiedene Kontextdimensionen, im Modell <i>Kontextarten</i> genannt, können durch verschiedene Ontologien strukturiert werden.
2b	Kombination separat strukturierter Datendimensionen	1	Es können nur auf bestimmten Ontologien basierende Objekte miteinander kombiniert werden, jedoch nicht verschiedene Ontologien in einem Objekt.
3a	Dynamische Elemente	4	Im Prinzip werden sämtliche Inhalte als dynamisch angenommen. Explizite Vorkehrungen für unterschiedlich dynamische Informationen fehlen allerdings.
3b	Generalisierende Ontologien	4	Zur Laufzeit können dynamisch neue Informationen hinzugefügt bzw. die Inhalte von existierenden Kontextvariablen aktualisiert werden. Dank ihrem hierarchischen Aufbau können Ontologien zur Laufzeit auch für existierende Anwendungen verständlich erweitert werden (vgl. [Kadn04b, S. 32 f]). Dies gilt allerdings nicht für Assoziationsarten.
3c	Darstellung der Historie von Kontextvariablen	2	Es wird keine Historie vorgesehen. Lediglich der letzte verfügbare Wert einer Kontextvariablen wird dargestellt.
4a	Datenqualitätsdeskriptoren	5	Es werden datenqualitätsorientierte Attribute vorgehalten, u. a. zur Beschreibung für deren Genauigkeit und Wahrscheinlichkeit (vgl. [Kadn04a, S. 40]).
4b	Datentypspez. Qualitätsdeskriptoren	5	Die Qualitätsattribute werden einzelnen Kontexten und damit Attributen zugeordnet und können für verschiedene Kontexte durchaus verschieden sein.
5	Schutz der Privatsphäre	2	Datenschutzaspekte werden nicht als Bestandteil des Kontextmodells, sondern von dessen Managementkomponente angesehen (vgl. [Kadn04a, S. 47]).
6a	Realweltobjektbezogene Struktur	5	Die Realwelt wird ausgehend von Realweltobjekten modelliert.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6b	Eineindeutige Realweltobjektmodelle	4	Realweltobjekte werden innerhalb des Kontextmodells eindeutig identifiziert, ein entgegengesetzter Link fehlt jedoch.
6c	Explizite Ontologie	4	Das Kontextmodell basiert auf expliziten Ontologien, allerdings wird nur eine Ontologie gleichzeitig unterstützt (vgl. [Kadn04a, S. 38 ff]).
6d	Darstellung von Kontextquellen	2	Externe Kontextquellen können nicht dargestellt werden.
6e	Inhaltliche Darstellung von Kontextvariablen	4	Externe Inhalte werden durch einen CIP in das Kontextmodell kopiert und stehen wertmäßig zur Verfügung. Dieser CIP erlaubt das automatisierte Zugreifen von Inhalten, jedoch nicht die automatische Erkennung veränderter Werte in den externen Kontextquellen.
6f	Verbindung vom Kontextmodell zu den Originaldaten	4	Externe Datensätze können durch dafür eingerichtete CIPs dargestellt werden, allerdings müssen für jeden Datensatz sämtliche Zugriffsinformationen komplett angegeben werden, da keine externen Datenquellen dargestellt werden können.
6g	Fusion von Kontextvariablen	1	Operationen sowie das Konzept synonymmer Kontextvariablen sind nicht bekannt.
6h	Integration von Methodik in das Datenmodell	1	Abgesehen von einer anfänglichen Aufbereitung der Sensorrohdaten zu Kontextinformationen in den CIPs kann keine weitere Methodik im Kontextmodell dargestellt werden.
6i	Darstellung des Datenpfades	1	Da keine Transformationen auf Kontextinformationen dargestellt werden können, kann auch kein Datenpfad dargestellt werden.
6j	Einschränkung der Sichtbarkeit von Konzepten	1	Die Sichtbarkeit einzelner Kontextobjekte kann nicht eingegrenzt werden.
6k	Synonyme Benennung von Topics	2	Objekte können keine synonymen Namen besitzen.
7a	Darstellung verschiedener Abstraktionsstufen	1	Verschiedene Abstraktionsstufen sind nicht vorgesehen.
7b	Endnutzertaugliche Programmiermethodiken	3	Endnutzer können zusätzliche Datenquellen angeben. Weitere Operationen sind aufgrund der fehlenden Einbettung von Methodik in das Kontextmodell nicht möglich.
7c	Problemspezifische Sichtweisen	1	Subjektive Sichtweisen auf das Modell können nicht erzeugt werden.

Anhang A.12 Gaia

Tabelle 38: Bewertung von *Gaja* hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	5	Sämtliche Modellelemente sind in der benötigten Form formalisiert.
1b	Modularität	1	Das Kontextmodell kann prinzipbedingt nicht aus einzelnen Teilen zusammengesetzt werden.
1c	Verteilte Darstellung und Speicherung	4	Da das Kontextmodell mittels des pervasiven Gaia-Frameworks realisiert wurde, kann es von verschiedenen Endgeräten zugegriffen werden, allerdings nur bei bestehender Online-Verbindung. Für eine funktionierende Replikation wäre auch das Kopieren der in Kap. 4.6.8 auf S. 128 beschriebenen Auswertungsmechanismen notwendig, die modellseitig jedoch nicht immer in einer replizierbaren Form vorgehalten werden.
1d	Unabhängigkeit von Programmiersprachen	5	Abfragen an das Expertensystem werden in <i>Prolog</i> formuliert, womit Client-Anwendungen im Prinzip in beliebigen Programmiersprachen verfasst sein können.
1e	Flexible Adressierungsmöglichkeiten	5	Assoziationen beliebiger Form lassen sich durch weitere Fakten darstellen.
2a	Separate Strukturierung von Datendimensionen	5	Verschiedene Kontextdimensionen können per Ontologie separat definiert werden.
2b	Kombination separat strukturierter Datendimensionen	4	Ein Objekt kann in Fakten aus verschiedenen Ontologien auftauchen, womit die Kombination separater Ontologien gegeben ist. Eine Zusammensetzung komplexer Datentypen aus separaten Teilontologien ist jedoch nicht möglich.
3a	Dynamische Elemente	4	Einzelne Provider können an Sensoren gekoppelt werden und so zur Erfassung schnell veränderlicher Kontextvariablen verwendet werden. Allerdings werden keine Attribute zur Beschreibung der Dynamik vorgesehen.
3b	Generalisierende Ontologien	5	Eine Erweiterung der verwendeten Ontologie zur Laufzeit ist möglich.
3c	Darstellung der Historie von Kontextvariablen	5	Einzelne Messwerte von Kontextvariablen werden mit einem Zeitstempel versehen in einer Datenbank persistent gemacht und stehen etwa für Analysezwecke zur Verfügung.
4	Datenqualität	2	Eine Behandlung unscharfer oder unterschiedlich wahrscheinlicher Aussagen über den Kontext ist im vorgestellten Modell nicht möglich. Erweiterungen zur Adressierung dieser Probleme werden vorgeschlagen, sind jedoch noch nicht realisiert (vgl. [RaCa03, S. 359, 361, 363]).
5	Schutz der Privatsphäre	2	Sicherheitsaspekte sind im gegenwärtigen Entwicklungsstand nicht implementiert, jedoch geplant (vgl. [RaCa03, S. 363]).
6a	Realweltobjektbezogene Struktur	1	Das Kontextmodell besteht lediglich aus einer Reihe von Aussagen über Realweltobjekte. Letztere werden nicht explizit als Modell dargestellt, sondern nur über Namen angesprochen, die jedoch nicht eindeutig sein müssen. Die einzelnen Informationen über Realweltobjekte werden verteilt in einzelnen Aussagen dargestellt.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6b	Eineindeutige Realweltobjektmodelle	1	Die Realweltobjekte werden lediglich über uneindeutige Identifikatoren wie Namen identifiziert.
6c	Explizite Ontologie	5	Die verwendeten Konzepte werden durch eine explizit formulierte Ontologie dargestellt und auf Validität überprüft. Damit ist das interne Kontextmodell erweiterbar, wenn die externe Ontologie erweitert wird.
6d	Darstellung von Kontextquellen	4	Externe Kontextquellen können durch <i>Context Provider</i> dargestellt werden, wobei ein <i>Context Provider</i> eine Reihe von unterschiedlichen Aussagen generieren kann. Allerdings ist den Aussagen später nicht mehr anzusehen, aus welcher Kontextquelle sie stammen.
6e	Inhaltliche Darstellung von Kontextvariablen	5	Inhalte aus externen Quellen werden mittels <i>Context Providern</i> in das Kontextmodell kopiert und dort persistent gespeichert.
6f	Verbindung vom Kontextmodell zu den Originaldaten	1	Es ist nicht möglich, die Statements einzelnen Kontextquellen oder gar einzelnen Datensätzen darin zuzuordnen.
6g	Fusion von Kontextvariablen	4	Aussagen können zwar miteinander kombiniert und so neue Erkenntnisse gewonnen werden, diese können jedoch nicht in die Datenbasis zurückgeschrieben, sondern nur direkt durch Client-Anwendungen verwendet werden.
6h	Integration von Methodik in das Datenmodell	4	Mittels <i>Interpretern</i> können Auswertungsroutinen für in der Datenbank enthaltene Kontextinformationen dargestellt werden, allerdings können deren Ergebnisse nicht in die Datenbank eingepflegt werden.
6i	Darstellung des Datenpfades	1	Da die Ergebnisse nicht in die Datenbasis zurückgeschrieben werden, können sie nicht in weiteren Operationen verwendet werden, wodurch kein Datenpfad entsteht.
6j	Einschränkung der Sichtbarkeit von Konzepten	1	Die gesamte Kontextdatenbasis ist global sichtbar.
6k	Synonyme Benennung von Topics	2	Realweltobjekte werden über Namen identifiziert. Es werden jedoch keine synonymen Namen für Realweltobjekte erwähnt, obwohl dies durchaus im Kontextmodell dargestellt werden könnte.
7a	Darstellung verschiedener Abstraktionsstufen	3	Im Kontextmodell selbst können nur <i>low level</i> -Informationen dargestellt werden. Diese können durch Interpreter transformiert und Client-Anwendungen zur Verfügung gestellt werden, allerdings nur in einem einzelnen Verarbeitungsschritt.
7b	Endnutzertaugliche Programmiermethodiken	1	Zur Benutzung des Modells ist die Kenntnis grundlegender mathematischer, insbesondere logischer und mengenorientierter Operationen notwendig, welche nicht ohne weiteres von einem durchschnittlichen Endnutzer erlernt werden können. Die Interpreter sind strukturell nicht in die Codebasis eingegliedert und verwenden ihrerseits abstrakte, endnutzeruntaugliche Prolog-Ausdrücke. Auch das Konzept der Zerlegung eines Entscheidungsalgorithmus in einzelne Regelmengen erfordert ein strukturiertes, nicht von Endnutzern realisierbares Vorgehen.
7c	Problemspezifisch Sichtweisen	2	Subjektive Sichtweisen auf Teile des Modells ließen sich auf Umwegen über synonyme Fakten und Regeln realisieren, dies ist jedoch nicht im beschriebenen Kontextmodell vorgesehen und auch nicht ohne weiteres vom eigentlichen Kontextmodell abgrenzbar.

Anhang A.13 Schmidt

Tabelle 39: Bewertung des tupelraumbasierten Kontextmodells von SCHMIDT hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	3	Das Kontextmodell definiert zunächst Kontextvariablen, keine Realweltobjekte oder Beziehungen zwischen diesen.
1b	Modularität	1	Da nur eine Ontologie verwendet wird, kann das Kontextmodell nicht aus verschiedenen Teilen zusammengesetzt werden.
1c	Verteilte Darstellung und Speicherung	4	Das Kontextmodell besteht aus einer Datenbasis sowie externen <i>Abstraktoren</i> oder <i>Operatoren</i> , die höherwertigen Kontext berechnen können. Für funktionsfähige Repliken müssen beide Dinge kopiert werden können.
1d	Unabhängigkeit von Programmiersprachen	5	Clientanwendungen formulieren Anfragen in einer speziellen Abfragesprache, wodurch sie in beliebigen Programmiersprachen verfasst sein können.
1e	Flexible Adressierungsmöglichkeiten	3	Verbindungen zwischen einzelnen Realweltobjekten können lediglich über eine zu schlußfolgernde Situation hergestellt werden, etwa dass mehrere Objekte zu einer Gruppe gehören. Dies ist jedoch ein eher umständliches Verfahren und nicht umfassend einsetz- oder miteinander kombinierbar.
2a	Separate Strukturierung von Datendimensionen	5	Separate Kontextdimensionen können in einem Tupelraum miteinander kombiniert werden.
2b	Kombination separat strukturierter Datendimensionen	4	Eine Situationsbeschreibung kann mittels verschiedener Ontologien definierte Attributen beinhalten. Diese Attribute können jedoch nicht zu Realweltrepräsentationen kombiniert werden.
3a	Dynamische Elemente	3	Kontextvariablen werden als dynamisch angenommen und deren Aktualität mittels modellseitig variabler Relevanzfunktionen modelliert (vgl. [ScGe01, S. 5]). Weitergehende Modellelemente zur Berücksichtigung der Dynamik von Kontextinformationen sind jedoch nicht vorgehen.
3b	Generalisierende Ontologien	3	Lediglich beim <i>MeLog</i> -Kontextmodell gibt es erweiterbare Ontologien, was modellseitig jedoch nicht wirklich vorgesehen ist.
3c	Darstellung der Historie von Kontextvariablen	5	Die Historie einer Kontextvariablen wird als explizite Modelldimension berücksichtigt. Einzelne Messwerte von Kontextvariablen werden als separate Tupel mit einem Zeitstempel versehen in einer Datenbank persistent gemacht.
4a	Datenqualitätsdeskriptoren	5	Der von SCHMIDT entwickelte <i>Fuzzyspace</i> kann von Kontextquellen gelieferte QoI-Attribute darstellen (vgl. [Schm02, S. 161]). Fernerhin wird die zeitliche und räumliche Nähe der einzelnen Kontexttupels zur Berechnung der Relevanz dieser Tupel hinsichtlich eines betrachteten Realweltausschnittes verwendet (vgl. [ScGe01, S. 5 f]).
4b	Datentypspez. Qualitätsdeskriptoren	2	Weitere datentypspezifische Qualitätsmetriken sind nicht vorgesehen.
5	Schutz der Privatsphäre	2	Sicherheitsaspekte sind im gegenwärtigen Entwicklungsstand nicht vorgesehen.
6a	Realweltobjektbezogene Struktur	3	Tupel stellen einzelne Messwerte von Realweltobjektmodellattributen dar. Trotzdem stellen Realweltobjektmodelle nicht das primäre Strukturierungskriterium in einem Tupelraum dar, vielmehr vereint ein Tupelraum sämtliche Messwerte unabhängig von den betreffenden Realweltobjekten.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
6b	Eineindeutige Realweltobjektmodelle	3	Die Realweltobjekte werden über eindeutige Identifikatoren dargestellt. Weitergehende Maßnahmen zur Erreichung von Eineindeutigkeit werden nicht getroffen.
6c	Explizite Ontologie	4	Die verwendeten Konzepte werden durch eine explizit formulierte Ontologie dargestellt, allerdings sind diese Ontologien nicht hierarchisch.
6d	Darstellung von Kontextquellen	3	Kontextquellen werden als <i>Kontextlieferanten</i> dargestellt (vgl. [ScGe01, S. 8]). Allerdings fehlen modellseitige Mechanismen zur Darstellung von Zugriffsverfahren bzw. -informationen.
6e	Inhaltliche Darstellung von Kontextvariablen	5	Inhalte werden mittels <i>Kontextlieferanten</i> in aktueller Form bereitgestellt.
6f	Verbindung vom Kontextmodell zu den Originaldaten	1	Informationen werden als reine Datentupel dargestellt, wobei keine Verknüpfung auf den externen Datensatz vorgesehen ist.
6g	Fusion von Kontextvariablen	2	Synonyme Kontextvariablen werden nicht explizit unterstützt.
6h	Integration von Methodik in das Datenmodell	4	Logik wird in Form von <i>Kontextabstraktoren</i> dargestellt (vgl. [Schm02, S. 177]). Diese stellen externe Agenten zur Analyse der Kontextinformationsbasis dar (vgl. 4.3.1 („Externe Agenten“, S. 108)).
6i	Darstellung des Datenpfades	4	Ein Datenpfad kann aufgrund der externen Abstraktoren nur schwierig dargestellt werden.
6j	Einschränkung der Sichtbarkeit von Konzepten	4	Die Sichtbarkeit von Kontextinformationen kann nur auf die räumliche oder zeitliche Umgebung von Referenzpunkten eingeschränkt werden. Beliebiger gestaltbare Eingrenzungen sind nicht möglich.
6k	Synonyme Benennung von Topics	1	Objekte werden gar nicht per Namen, sondern nur per ID identifiziert.
7a	Darstellung verschiedener Abstraktionsstufen	3	Es werden zwei Abstraktionsstufen unterschieden, die Ebene der <i>low level</i> -Kontextvariablen und sowie daraus abgeleitete Situationsinformationen (vgl. Abb. 33 auf Seite 99). Weitergehende, flexiblere Mechanismen existieren nicht.
7b	Endnutzertauglich Programmiermethodiken	3	Endnutzer können die Modellinformationen einsehen sowie Abstraktoren anpassen. Dies ist jedoch nicht im speziellen endnutzertauglich gestaltet.
7c	Problemspezifische Sichtweisen	1	Verschiedene Sichtweisen auf die Datenbasis werden nicht unterstützt.

Anhang A.14 Context Maps

Tabelle 40: Bewertung des Kontextmodells des Owl Context Service hinsichtlich der Eignung zur Integration heterogener kontextbezogener Informationen

Nr	Name	Pt	Begründung
1a	Formalisierungsgrad	5	Die hierarchische Definition sowohl von Topics als auch Assoziationen mittels Ontologien ist möglich.
1b	Modularität	5	Ein Kontextmodell kann aus mehreren Maps zusammengesetzt werden, wobei Informationen in beliebigen Maps adressiert werden können. Darüber hinaus ist mittels <i>Abstrakter Assoziationen</i> die separate und wiederverwendbare Definition abgegrenzter Teile des Kontextmodells möglich.
1c	Verteilte Darstellung und Speicherung	5	Da das Kontextmodell sowohl die betreffenden Daten, Ontologien als auch Methoden enthält, ist es in einer voll funktionsfähigen Form auf andere Geräte replizierbar.
1d	Unabhängigkeit von Programmiersprachen	5	Um das Modell zugreifen und verwenden zu können, ist keine besondere Programmiersprache notwendig, da die modellinternen Methoden auch modellintern realisiert und ausgeführt werden.
1e	Flexible Adressierungsmöglichkeiten	5	Context Maps erlauben nicht nur die Angabe beliebiger Assoziationen zwischen Realweltobjektmodellen, sondern auch deren besonders flexible und selbst wieder kontextabhängige Gestaltung mittels <i>Aktiver Assoziationen</i> .
2a	Separate Strukturierung von Datendimensionen	5	Da ein Modell beliebig viele Ontologien miteinander kombinieren kann, können verschiedene Datendimensionen separat modelliert und global verfügbar gemacht werden.
2b	Kombination separat strukturierter Datendimensionen	5	Da multiple Vererbung unterstützt wird, können konkrete Realweltobjekte Attribute und Klassenbezeichnungen von mehreren Superklassen aus verschiedenen Ontologien implementieren und daher deren Rolle einnehmen.
3a	Dynamische Elemente	5	Context Maps unterstützen sowohl den Zugriff auf externe Kontextvariablen als auch deren Zwischenspeicherung im Modell. Die Dynamik des Zugriffs auf externe Kontextquellen kann mittels der Dynamikdeskriptoren <i>MinTimeBetweenUpdates</i> und <i>MaxTimeBetweenUpdates</i> für jedes Attribut separat geregelt werden.
3b	Generalisierende Ontologien	5	Die verwendeten Ontologien für Realweltobjektclassen und -assoziationen sind flexibel erweiter- und miteinander kombinierbar.
3c	Darstellung der Historie von Kontextvariablen	5	Die Historie von Kontextvariablen, d. h. Attributen von Realweltobjektrepräsentationen erfolgt in einer expliziten Form, welche in gleicher Weise wie deren aktueller Inhalt zugegriffen werden kann.
4	Datenqualität	5	Zur Beschreibung der Datenqualität der Kontextvariablen wird das von <i>Wang</i> und <i>Henricksen</i> entwickelte Datenqualitätsmodell verwendet, welches sämtliche in diesem Zusammenhang entstehenden Anforderungen erfüllt.
5a	Datenschutzdeskriptoren	5	Zugriffsrechte können mittels Sicherheitsdeskriptoren für beliebig granulare Modellelemente festgelegt werden. Diese Deskriptoren stellen ein explizites Element des Metamodells dar, wodurch deren einheitliche Verwendung und Interpretation für jede Context Map Engine festgelegt wird.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
5b	Darstellung von Akteuren	4	Informationen über zugreifende Instanzen stellen selbst Kontextinformationen dar, welche durch Context Maps dargestellt werden können. Aufgrund verschiedenen Ansätze und Verfahren werden an dieser Stelle jedoch keine weitergehenden Festlegungen getroffen, sind modellseitig aber vorgesehen und können durch zusätzliche Kontextinformationen und Sicherheitsdeskriptoren akkurat implementiert werden.
5c	Vertrauensstufen	4	Die Festlegung verschiedener Vertrauensstufen wäre durch entsprechende Deskriptoren möglich. Eine konkrete Festlegung soll an dieser Stelle jedoch nicht vorgenommen werden, da dies den zu unterschiedlichen Anforderungen in der Realität nicht gerecht werden würde.
6a	Realweltobjekt-bezogene Struktur	5	Nicht nur Realweltobjekte, sondern auch beliebige Konzepte aus der Realwelt können durch <i>Topics</i> dargestellt werden, zu welchen beliebige Kontextvariablen in Form von <i>Attributen</i> zugeordnet werden können.
6b	Eineindeutige Realweltobjektmodelle	5	Die zur Identifizierung von <i>Topics</i> verwendeten <i>Subject Indicators</i> , welche vor allem aus einer URI bestehen, stellen ein weltweit eineindeutiges Identifikationsschema dar. Zusätzlich können diese in Form von <i>Published Subject Indicators</i> weltweit eindeutig publiziert und verwendet werden.
6c	Explizite Ontologie	5	Auf der <i>Ontologieschicht</i> werden die von Context Maps verwendeten Ontologien in einer expliziten Form dargestellt. Dabei wird angeregt, Konzepte aus diesen expliziten Ontologien in mehreren Anwendungen wiederzuverwenden, um die Integration dieser Informationen weiter zu erleichtern.
6d	Darstellung von Kontextquellen	5	Externe Kontextquellen werden im Kontextmodell als separate Realweltobjekte dargestellt, zu denen sämtliche benötigten Zugangsinformationen, u. a. der Typ der Kontextquelle dargestellt werden. Aus diesem Typ wird ein geeignetes Zugriffsmodul abgeleitet und mit den Logindaten konfiguriert.
6e	Inhaltliche Darstellung von Kontextvariablen	5	Sowohl das Auslesen externer Kontextvariablen als auch die Darstellung dieser Inhalte im Kontextmodell sind möglich, wobei entsprechende Synchronisationen durch die Dynamikdeskriptoren des entsprechenden Attributes konfiguriert werden.
6f	Verbindung vom Kontextmodell zu den Originaldaten	5	Die Zuordnung der modellinternen Kontextvariableninhalte zu ihren korrespondierenden Werten in den Kontextquellen wird in der Beschaffungsmatrix eines Realweltobjektattributes hergestellt. Durch diesen expliziten Zusammenhang ist auch eine Aktualisierung in beiden Richtungen möglich.
6g	Fusion von Kontextvariablen	5	Zur Kombination synonyme Kontextvariablen aus verschiedenen Kontextquellen dienen die <i>Beschaffungsmatrix</i> von Attributen. Dort können nicht nur die entsprechenden Kontextvariablen identifiziert und lokalisiert werden, sondern auch die entsprechenden Transformations- und Integrationsschritte mittels <i>Aktiver Assoziationen</i> modelliert werden.
6h	Integration von Methodik in das Datenmodell	5	Sämtliche der in diesem Punkt zusammengefassten Anforderungen werden durch <i>Aktive Assoziationen</i> erfüllt.
6i	Darstellung des Datenpfades	5	Der Datenpfad wird durch Context Maps in einer im Vergleich zu anderen Verfahren wesentlich plastischeren Form abgebildet. So sind nicht nur die Quellen und daraus entnommenen Originaldaten sowie die daraus abgeleiteten Kontextvariablen miteinander verkettet sichtbar, sondern u. U. auch markante Zwischenschritte der Transformation. Dadurch können diese Informationen auch in mehreren Operationen wiederverwendet werden. Indem <i>Topics</i> bestimmte <i>Member Role</i> genannte Rollen in Aktiven Assoziationen spielen können, wird nicht nur die Aktualisierungsabhängigkeit selbst, sondern auch die Rolle der einzelnen Attribute in diesem Zusammenhang deutlich.
6j	Einschränkung der Sichtbarkeit von Konzepten	5	Innerhalb der Informationsbasis können mit den von Topic Maps bekannten <i>Scopes</i> beliebige Bereiche definiert werden, auf welche die Sichtbarkeit bestimmter <i>Topics</i> beschränkt werden kann.
6k	Synonyme Benennung von Topics	5	<i>Topics</i> können mittels der von Topic Maps bekannten <i>Variant Names</i> synonym benannt werden.

Fortsetzung auf der folgenden Seite ...

Nr	Name	Pt	Begründung
7a	Darstellung verschiedener Abstraktionsstu- fen	5	Ganze Maps können durch eine hierarchische Anordnung in verschiedenen Abstraktionsstufen dargestellt werden. Mittels <i>Abstrakter Assoziationen</i> können zusätzlich einzelne Teile des Kontextmodells in verschiedenen Abstraktionsebenen dargestellt werden.
7b	Endnutzertaugliche Programmierme- thodiken	4	Obwohl das Kontextmodell sicher nicht als ausgereift endnutzertauglich bezeichnet werden kann, stellt es doch wichtige Schritte in diese Richtung dar. So wird die Komplexität der Problemdomäne entscheidend reduziert und das Modell weist eine inzwischen jedem Endnutzer durch das Internet bekannte Netztopologie auf, die sich im Vergleich zu anderen Verfahren durchaus endnutzertauglich visualisieren ließe. Auch die Mittel von Abstrakten Assoziationen können zu einer deutlichen Reduzierung der dargestellten Informationsmenge führen.
7c	Problemspezifisch Sichtweisen	4	Beliebige Teile eines Informationsmodells können in separaten Maps, welche auf einer anderen Ontologie basieren und ihre Informationen komplett aus dem bestehenden Kontextmodell beziehen, in einer anderen, d. h. anwendungs- oder problemspezifischen Ontologien dargestellt werden. Allerdings wird dieser Vorgang derzeit nicht systematisch unterstützt, so dass diese Ontologietransformationen manuell durchgeführt werden müssen.

Anhang B XML-Notation für Context Maps

Die Struktur von Context Maps kann abgesehen von Aktiven Assoziationen relativ geradlinig als XML dargestellt werden. Sie basiert auf dem in [PeMo01] spezifizierten XML-basierten Topic-Map-Standard, d. h. an dieser Stelle werden nur die Änderungen gegenüber diesem Standard angegeben.

Anhang B.1 Context Map

Abb. 122 zeigt die Notation für eine gesamte Context Map. Die fehlenden Elemente werden später zu gegebener Zeit definiert.

```
<!DOCTYPE contextMap [
  <!ELEMENT contextMap (description?,
                        (realworldUniverse | realworldUniverseRef),
                        instances?,
                        (electronicUniverse | electronicUniverseRef)?)>
]>
```

Abbildung 122: DTD der gesamten Context Map

Anhang B.2 Realweltuniversum

Das gesamte Realweltuniversum ist im Tag `RealworldUniverse` enthalten. Es hat die in Abb. 123 dargestellte Struktur. Statt konkreter Elemente können auch Referenzen auf an einer anderen Stelle existierende Elemente verwendet werden. Dies wird durch Verwendung von `elementRef` anstatt von `element` erreicht, wobei für Element die Universenelemente `realworldUniverse`, `ontologyMap` bzw. `sceneMap` verwendet werden können.

```
<!ELEMENT realworldUniverse (description?, ontologyMaps, sceneMaps)>
<!ATTLIST realworldUniverse id ID #required
                        version PCDATA #implied>
  <!ELEMENT ontologyMaps((map | mapRef)+)>
  <!ELEMENT sceneMaps((map | mapRef)+)>
  <!ELEMENT realworldUniverseRef (EMPTY)>
  <!ATTLIST realworldUniverseRef xlink:href CDATA #REQUIRED>
```

Abbildung 123: DTD eines Realweltuniversums

Anhang B.3 Instanzen

Das Element `Instances` enthält die konkreten Realweltmodelle. Abb. 124 auf Seite LXVIII zeigt den entsprechenden Teil aus der DTD.

```
<!ELEMENT instances (map*)>
```

Abbildung 124: DTD einer *Instance-Map*

Anhang B.4 Kontextquellenebene

Das Element `ContextSourceTypes` enthält die Definitionen der Kontextquellentypen. Abb. 125 zeigt den entsprechenden Teil aus der DTD.

```
<!ELEMENT electronicUniverse(description?, contextSource*)
<!ELEMENT contextSource (realworldObjectType, uriSchema, accessOperation)>
<!ELEMENT realworldObjectType (topicRef)>
<!ELEMENT uriSchema (#PCDATA)>
<!ELEMENT accessOperation (name, description?, parameter*, result)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT description (#PCDATA)>
<!ELEMENT parameter (name, type)>
<!ELEMENT type (#PCDATA)>
<!ELEMENT result (type)>
```

Abbildung 125: DTD von Kontextquellentypen

Anhang B.5 Allgemeine Context-Map-Elemente

Anhang B.5.1 Maps

Anhang B.5.2 Topic

Abb. 127 zeigt die DTD eines Topics.

Anhang B.5.3 Attribute

Abb. 128 auf Seite LXIX zeigt die DTD eines Attributes.

```
<!ELEMENT map (description?, subjectIdentity?,
               (topic | association | abstractAssociation | activeAssociation)*)>
  <!ATTLIST map id ID #REQUIRED
               instanceOf xlink:href #IMPLIED>
<!ELEMENT mapRef (EMPTY)>
  <!ATTLIST mapRef xlink:href CDATA #REQUIRED>
```

Abbildung 126: DTD einer Map


```

<!ELEMENT topic (instanceOf*, subclassOf*, subjectIdentity?,
                baseName, description?, attribute*)>
  <!ATTLIST topic id ID #required
                xlink:href CDATA #IMPLIED>

```

Abbildung 127: DTD eines Topics

```

<!ELEMENT attribute (basename, description?, attributeValue, accessMap?,
                    history?, quality?, security?, dynamics?)>
  <!ATTLIST attribute id ID #required> \codeSkip
<!ELEMENT attributeValue (standardizedValue | topicRef)>
  <!ELEMENT standardizedValue (unit?, value)>
    <!ELEMENT unit (#PCDATA)>
    <!ELEMENT value (#PCDATA)> \codeSkip
<!ELEMENT accessMap (map)> \codeSkip
<!ELEMENT history (historicalValue*)>
  <!ELEMENT historicalValue (attributeValue, quality?)>
    <!ATTLIST historicalValue timeStamp PCDATA #required> \codeSkip
<!ELEMENT quality (qualityDescriptor*)>
  <!ELEMENT qualityDescriptor (qualityParameter, qualityMetrics)>
    <!ELEMENT qualityParameter (topicRef)>
    <!ELEMENT qualityMetrics (StandardizedValue)> \codeSkip
<!ELEMENT security (securityDescriptor*)>
  <!ELEMENT securityDescriptor (actor)>
    <!ATTLIST securityDescriptor type (accessible | copyable | fusable) #required>
    <!ELEMENT actor (topicRef)> \codeSkip
<!ELEMENT dynamics (dynamicsDescriptor*)>
  <!ELEMENT dynamicsDescriptor (minTimeBetweenUpdates, maxTimeBetweenUpdates)>
    <!ELEMENT minTimeBetweenUpdates (StandardizedValue)>
    <!ELEMENT maxTimeBetweenUpdates (StandardizedValue)>

```

Abbildung 128: DTD eines Attributes

Anhang B.5.4 Normale Assoziationen

Normale Assoziationen können zusätzlich zu den durch den Topic-Map-Standard definierten Merkmalen verschiedene Namen erhalten. Dies ist notwendig, um den Anforderungen bei der Integration heterogener Informationsbestände gerecht zu werden.

```

<!ELEMENT association (instanceOf?, subclassOf?, baseName?,
                     description?, scope?, member+)>
  <!ATTLIST association id ID #required>
  <!ELEMENT member (topicRef)>

```

Abbildung 129: DTD einer Normalen Assoziation

Anhang B.5.5 Abstrakte Assoziationen

Abb. 130 zeigt die Definition von Abstrakten Assoziationen.

```
<!ELEMENT abstractAssociation (instanceOf?, subclassOf?, baseName?,
                               description?, scope?, member+, lowLevelMap)>
<!ATTLIST abstractAssociation id ID #required>
<!ELEMENT lowLevelMap (map)>
```

Abbildung 130: DTD einer *Abstrakten Assoziation*

Anhang B.5.6 Aktive Assoziationen

Abb. 131 zeigt die Definition von Aktiven Assoziationen.

```
<!ELEMENT activeAssociation (instanceOf?, subclassOf?, baseName?,
                              description?, scope?, member+, codeBlock)>
<!ATTLIST activeAssociation id ID #required>
<!ELEMENT codeBlock (#PCDATA)>
  <!ATTLIST codeBlock language (#PCDATA) #required>
```

Abbildung 131: DTD einer *Aktiven Assoziation*

```
(1) [Quellentyp]://[Logindaten]@[Quellenadresse]?[Lokalisierung des Datensatzes]
(2) [Quellentyp]://[Logindaten zur Quelle]@[Adresse der Quelle]
(3) [Quellenname]?[Ort des Datensatzes in der Quelle]
```

Abbildung 132: Format der erweiterten URI-Syntax

Anhang C URI-Notation für Occurrences

Die Identifikation von Datensätzen in externen Datenquellen wird mittels der konsequenten Verwendung der durch den URI-Standard bereitgestellten Mittel ermöglicht und folgt dem bei URI üblichen Aufbau, der in Abb. 132 in Zeile (1) dargestellt ist. Das `Quellentyp` genannte URI-Schema spezifiziert dabei die Art der Quelle, woraus sich das zum Zugriff auf die betreffende Quelle zu verwendende Verfahren ableitet. Die Logindaten sind ebenfalls quellentypspezifisch, sehr oft kann hierfür aber das aus dem URI-Standard bekannte Schema `Nutzername:Passwort` verwendet werden. Die Adresse der Quelle wird i. d. R. in Form der IP-Adresse des Gerätes angegeben, auf welcher die betreffende Anwendung läuft. Das Format der `Lokalisierung des Datensatzes` genannten Adressierung der gesuchten Kontextvariablen innerhalb der Quelle hängt ebenfalls von der Art der Datenquelle ab.

Die URI kann in zwei Formen angegeben werden, zum einen in einer kompletten Form wie in Zeile (1) in Abb. 132 dargestellt, zum anderen können die Informationen über die Datenquelle getrennt von den Informationen über einen einzelnen Datensatz angegeben werden, wie in den Zeilen (2) und (3) dargestellt. Diese getrennte Definition von Quellen und darin befindlichen Datensätzen ermöglicht die Wiederverwendung einer Quellendefinition zur Adressierung mehrerer Datensätze.

Es folgen einige Beispiele. Zeile (1) in Abb. 133 stellt dar, wie der Vorname der Person *John Doe* aus dem Mitarbeiterverzeichnis der TU-Dresden adressiert wird. Dabei gibt das URI-Schema `mysql` den Typ der Datenquelle, in diesem Fall eine relationale Datenbank, an, so dass automatisch ein geeigneter Wrapper zum Zugriff ausgewählt werden kann. Darauf folgend werden die Logindaten zum Zugriff auf die Datenbank angegeben. Die Adresse der Kontextquelle ist `tu-dresden.de/staff`, der Vorname von John wird darin mittels der Query `query=SELECT vorname FROM person WHERE id=123` adressiert.

```
(1) mysql://user:pass@db.tu-dresden.de/staff?query=SELECT vorname FROM person
    WHERE id=123
(2) appleAddressbook://jeff:secret@141.76.44.101?entry=Jeff Malone&field=First
(3) MaloneAddressbook: appleAddressbook://jeff:secret@141.76.44.101
(4) MaloneAddressbook?entry=Jeff Malone&field=First
```

Abbildung 133: Beispiele für die direkte Adressierung eines Datensatzes mittel der erweiterten URI-Syntax

Gleiches gilt für die in der zweiten Zeile dargestellte Adressierung des Vornamens von *Jeff Malone* aus der Adressbuchanwendung auf dem durch die angegebene IP-Adresse spezifizierten Gerät, wobei ebenfalls Logininformationen für diese Quelle angegeben sind. Die Adressierung der gewünschten Infor-

mation erfolgt hier in zwei Schritten, zunächst wird der betreffende Datensatz über den Parameter `entry` identifiziert, daraufhin das gewünschte Datenfeld in diesem Datensatz über den Parameter `field`. Die Zeilen (3) und (4) stellen die selbe URI in der zerlegten Form dar, wobei Zeile (3) die Quelle, welcher der Name „MaloneAddressbook“ gegeben wird und Zeile (4) einen Datensatz in der Quelle adressieren.

Anhang D Regeln zur Wohlgeformtheit von Context Maps

allgemein

- zirkulare Abhängigkeiten zwischen Aktiven Assoziationen sind nicht erlaubt,
- es darf keine abgegrenzten „Inseln“ im Kontextmodell geben, die über keine Assoziation mit dem restlichen Kontextmodell verbunden sind.

Universen

- sämtliche Topics müssen ontologisch definiert sein,
- sämtliche ontologischen Informationen müssen in einer *Ontology Map* vorgenommen werden,
- eine Ontology-Map darf nur ontologische Informationen, d. h. nur Topics, die Realweltobjektclassen repräsentieren sowie Assoziationen vom Typ *superclass_subclass* enthalten,
- szenische Beschreibungen sind optional,
- jede Rolle darf nur einmal in einer Szene auftauchen,
- eine Instanz einer Rolle muß sämtliche durch die Rolle definierten Attributtypen implementieren.

Universentransformation

- Sämtliche Informationen dürfen nur aus dem zu konvertierenden Universum entnommen werden.

Kontextquellen

- Für jede Kontextquellenrepräsentation muß ein entsprechendes Kontextquellentypmodell auf Ebene 3b existieren,
- Die Art und Weise des Zugriffs auf eine Kontextquelle muß von einem Kontextquellentypmodell auf Ebene 3b und nicht in einer Aktiven Assoziation auf Ebene 2 definiert werden.

Beschaffungsmap

- es darf nur ein Ergebnistopic geben,
- es muß mindestens eine Aktive Assoziation geben, die den entsprechenden Attributwert im Ergebnistopic setzt.

Anhang E Anwendungskontextmodelle

An dieser Stelle wird das Kontextmodell der in Kap. 6.2 („Funktionale Validierung“, S. 175) dargestellten Anwendungen abgebildet.

Anhang E.1 Generische Grundlage

Sämtliche Kontextmodelle basieren auf einer generischen Grundlage, welche die grundlegenden Realweltobjekttypen in einer wiederverwendbaren Form definiert. Diese ist im folgenden dargestellt und unter der URL <http://www.cm.org/standards/general.universe.cm> zugreifbar.

Anhang E.1.1 Ontologien

```
<contextMap>
  <realworldUniverse id="General_Real_World_Objects">
    <ontologyMaps>
      <map>

        <!-- Definition von "Realweltobjekt" -->
        <topic id="cRealworldObject">
          <instanceOf>
            <subjectIndicatorRef xlink:href="http://www.topicmaps.org/xtm/1.0/core.xtm#class">
            </instanceOf>
          <baseName>
            <baseNameString>Real world object</baseNameString>
          </baseName>
          <description>
            An arbitrary real world object.
            This class is used as the base class for all real world object classes.
          </description>
        </topic>

        <!-- Definition der Klasse "Mensch" als Subklasse von "Realweltobjekt" -->
        <topic id="cHuman">
          <subclassOf>
            <topicRef xlink:href="#cRealworldObject">
            </subclassOf>
          <baseName>
            <baseNameString>Human</baseNameString>
            <variant> <!-- deutscher Name von "Human" -->
              <parameters>
                <subjectIndicatorRef
                  xlink:href="http://www.topicmaps.org/xtm/1.0/country.xtm#de">
                </parameters>
                <resourceData>Mensch</resourceData>
              </variant>
            </baseName>
            <description>A human.</description>
          </topic>

        <!-- Definition der Klasse "Gerät" als Subklasse von "Realweltobjekt" -->
        <topic id="cDevice">
          <subclassOf>
            <topicRef xlink:href="#cRealworldObject">
            </subclassOf>
          <baseName>
            <baseNameString>Device</baseNameString>
          </baseName>
          <description>Any type of hardware.</description>
        </topic>

        <!-- Definition der Klasse "Anwendung" als Subklasse von "Realweltobjekt" -->
        <topic id="cApplication">
          <subclassOf>
            <topicRef xlink:href="#cRealworldObject">

```

```

    </subclassOf>
    <baseName>
      <baseNameString>Application</baseNameString>
    </baseName>
    <description>Any type of software.</description>
  </topic>

  <!-- Definition der Klasse "Kontextquelle" als Subklasse von "Realweltobjekt" -->
  <topic id="cContextSource">
    <subclassOf>
      <topicRef xlink:href="#cRealworldObject">
    </subclassOf>
    <baseName>
      <baseNameString>Context Source</baseNameString>
    </baseName>
    <description>A source of contextual information.</description>
  </topic>

  <!-- Definition der Klasse "Dokument" als Subklasse von "Realweltobjekt" -->
  <topic id="cDocument">
    <subclassOf>
      <topicRef xlink:href="#cRealworldObject">
    </subclassOf>
    <baseName>
      <baseNameString>Document</baseNameString>
    </baseName>
    <description>Any type of document.</description>
  </topic>

  <!-- Definition der Klasse "Company" als Subklasse von "Realweltobjekt" -->
  <topic id="cCompany">
    <subclassOf>
      <topicRef xlink:href="#cRealworldObject">
    </subclassOf>
    <baseName>
      <baseNameString>Company</baseNameString>
    </baseName>
    <description>A company.</description>
  </topic>

  <!-- Definition der Klasse "Profil" als Subklasse von "Dokument" -->
  <topic id="cProfile">
    <subclassOf>
      <topicRef xlink:href="#cDocument">
    </subclassOf>
    <baseName>
      <baseNameString>Profile</baseNameString>
    </baseName>
    <description>Information about something.</description>
  </topic>

</map>
</ontologyMaps>

```

Anhang E.1.2 Szenen

Innerhalb des eben beschriebenen generischen Realweltuniversums definiert die folgende Szene in ebenfalls noch sehr allgemeiner Form die typischerweise zwischen diesen elementaren Realweltobjekttypen möglichen Verbindungen.

```

<sceneMaps>
  <map>

    <!-- Rolle "Person" -->
    <topic id="rPerson">
      <instanceOf>
        <topicRef xlink:href="#cHuman" />
      </instanceOf>
    </topic>

```

```
<!-- Rolle "Gerät" -->
<topic id="rDevice">
  <instanceOf>
    <topicRef xlink:href="#cDevice" />
  </instanceOf>
</topic>

<!-- Rolle "Anwendung" -->
<topic id="rApplication">
  <instanceOf>
    <topicRef xlink:href="#cApplication" />
  </instanceOf>
</topic>

<!-- Rolle "Kontextquelle" -->
<topic id="rContextSource">
  <instanceOf>
    <topicRef xlink:href="#cContextSource" />
  </instanceOf>
</topic>

<!-- Rolle "Dokument" -->
<topic id="rDocument">
  <instanceOf>
    <topicRef xlink:href="#cDocument" />
  </instanceOf>
</topic>

<!-- Rolle "Firma" -->
<topic id="rCompany">
  <instanceOf>
    <topicRef xlink:href="#cCompany" />
  </instanceOf>
</topic>

<!-- Rolle "Profil" -->
<topic id="rProfile">
  <instanceOf>
    <topicRef xlink:href="#cProfile" />
  </instanceOf>
</topic>

<!-- Definition der Assoziationen zwischen den Rollen -->

<!-- Person benutzt Gerät -->
<association id="person_uses_device">
  <baseName>
    <baseNameString>Person uses device</baseNameString>
  </baseName>
  <member>
    <topicRef xlink:href="#rPerson"/>
  </member>
  <member>
    <topicRef xlink:href="#rDevice"/>
  </member>
</association>

<!-- Person benutzt Anwendung -->
<association id="person_uses_application">
  <baseName>
    <baseNameString>person uses application</baseNameString>
  </baseName>
  <member>
    <topicRef xlink:href="#rPerson"/>
  </member>
  <member>
    <topicRef xlink:href="#rApplication"/>
  </member>
</association>
```



```

<!-- Eine Person hat einen Vertrag mit einer Firma -->
<association id="person_has_contract_with_company">
  <baseName>
    <baseNameString>person has a contract with company</baseNameString>
  </baseName>
  <member>
    <topicRef xlink:href="#rPerson"/>
  </member>
  <member>
    <topicRef xlink:href="#rContract"/>
  </member>
  <member>
    <topicRef xlink:href="#rCompany"/>
  </member>
</association>

<!-- Eine Person hat ein Profil -->
<association id="person_has_profile">
  <baseName>
    <baseNameString>person has profile</baseNameString>
  </baseName>
  <member>
    <topicRef xlink:href="#rPerson"/>
  </member>
  <member>
    <topicRef xlink:href="#rProfile"/>
  </member>
</association>

<!-- Anwendung läuft auf Gerät -->
<association id="application_on_device">
  <baseName>
    <baseNameString>application runs on device</baseNameString>
  </baseName>
  <member>
    <topicRef xlink:href="#rApplication"/>
  </member>
  <member>
    <topicRef xlink:href="#rDevice"/>
  </member>
</association>

</map>
</sceneMaps>
</realworldUniverse>
</contextMap>

```

Anhang E.2 Nutzerprofil

Das Realweltuniversum des verwendeten Nutzerprofils wird ebenfalls an einer externen Stelle definiert, damit es von mehreren Nutzern wiederverwendet werden kann. Es ist unter der URL <http://www.cm.org/standards/userprofile.cm> zugreifbar.

Anhang E.2.1 Ontologien

```

<contextMap>
  <realworldUniverse id="UserProfile" version="1">
    <ontologyMaps>
      <map>
        <!-- Definition von Terminplaner -->
        <topic id="cDiary">
          <subclassOf>
            <topicRef xlink:href="http://www.cm.org/maps/general.ontology.cm#cApplication">
          </subclassOf>
          <baseName>

```

```

    <baseNameString>Diary</baseNameString>
    <variant>
      <resourceData>Scheduler</resourceData>
    </variant>
  </baseName>
  <description>
    A scheduling application, contains the appointments of its user.
  </description>
</topic>

<!-- Definition von Messenger -->
<topic id="cMessenger">
  <subclassOf>
    <topicRef xlink:href="http://www.cm.org/maps/general_ontology.cm#cApplication">
  </subclassOf>
  <baseName>
    <baseNameString>Messenger</baseNameString>
  </baseName>
  <description>
    An application that allows its user to communicate.
  </description>
</topic>

<!-- Definition von Adressbuch -->
<topic id="cAddressbook">
  <subclassOf>
    <topicRef xlink:href="http://www.cm.org/maps/general_ontology.cm#cApplication">
  </subclassOf>
  <baseName>
    <baseNameString>Address Book</baseNameString>
  </baseName>
  <description>
    An application that contains the address of people its user knows.
  </description>
</topic>

<!-- Definition von Termin -->
<topic id="cAppointment">
  <subclassOf>
    <topicRef xlink:href="http://www.cm.org/maps/general_ontology.cm#cDocument">
  </subclassOf>
  <baseName>
    <baseNameString>Appointment</baseNameString>
  </baseName>
  <description>An appointment of the user inside a diary.</description>
</topic>

<!-- Definition von Message -->
<topic id="cMessage">
  <subclassOf>
    <topicRef xlink:href="http://www.cm.org/maps/general_ontology.cm#cDocument">
  </subclassOf>
  <baseName>
    <baseNameString>Message</baseNameString>
  </baseName>
  <description>
    A textual message from or to the user, stored in a messenger application.
  </description>
</topic>

<!-- Definition von Kontaktperson -->
<topic id="cAcquaintance">
  <subclassOf>
    <topicRef xlink:href="http://www.cm.org/maps/general_ontology.cm#cDocument">
  </subclassOf>
  <baseName>
    <baseNameString>aquaintance</baseNameString>
    <variant>
      <resourceData>contact</resourceData>
    </variant>
  </baseName>

```

```

    <description>
      Information about a person, stored in an address book.
    </description>
  </topic>

  <!-- Definition von Mobilfunkvertrag -->
  <topic id="cMobileContract">
    <subclassOf>
      <topicRef xlink:href="http://www.cm.org/maps/general_ontology.cm#cDocument">
    </subclassOf>
    <baseName>
      <baseNameString>Mobile Contract</baseNameString>
    </baseName>
    <description>
      A subscription of the user for mobile services.
    </description>
  </topic>

  <!-- Definition von Mobilfunkprovider -->
  <topic id="cMobileProvider">
    <subclassOf>
      <topicRef xlink:href="http://www.cm.org/maps/general_ontology.cm#cCompany">
    </subclassOf>
    <baseName>
      <baseNameString>Mobile Provider</baseNameString>
    </baseName>
    <description>
      A company that provides mobile services to users.
    </description>
  </topic>

  <!-- Definition von Speisepräferenzen -->
  <topic id="cFoodPreferences">
    <subclassOf>
      <topicRef xlink:href="http://www.cm.org/maps/general_ontology.cm#cProfile">
    </subclassOf>
    <baseName>
      <baseNameString>food preferences</baseNameString>
    </baseName>
    <description>
      Information about the type of food the user prefers.
    </description>
  </topic>

  <!-- Definition von Stimmprofil -->
  <topic id="cVoiceProfile">
    <subclassOf>
      <topicRef xlink:href="http://www.cm.org/maps/general_ontology.cm#cProfile">
    </subclassOf>
    <baseName>
      <baseNameString>voice profile</baseNameString>
    </baseName>
    <description>
      Information about the characteristics of the user's voice.
    </description>
  </topic>

  </map>
</ontologyMaps>

```

Anhang E.2.2 Szenen

Zu dieser Ontologie gibt es eine analoge Beschreibung der zwischen diesen Objekten bestehenden Verhältnisse in einem typischen Nutzerprofil. Diese Informationen werden durch die im folgenden dargestellte Szene repräsentiert.

```

<sceneMaps>
  <map id="sUserProfile">

    <!-- Definition der Rolle "Nutzer" -->

```

```

<topic id="rUser">
  <baseName>
    <baseNameString>User</baseNameString>
    <variant>
      <parameters>
        <subjectIndicatorRef
          xlink:href="http://www.topicmaps.org/xtm/1.0/country.xtm#de">
        </parameters>
        <resourceData>Nutzer</resourceData>
      </variant>
    </baseName>
    <description>The person that is described by this user profile</description>
    <instanceOf>
      <topicRef xlink:href="#cHuman" />
    </instanceOf>
    <attribute>
      <baseName>
        <baseNameString>Location</baseNameString>
      </baseName>
      <description>
        The coordinates of the current location of the user.
        The value is determined using the information within this user profile.
      </description>
      <accessMap>
        <map> <!-- Accessmap for Attribute "location" of role "user" -->

        <!-- Link to the user -->
        <topic id="user"
          xlink:href="#rUser" />
        <!-- dummy topics, are filled later using active associations -->
        <topic id="rMobileDevice">
          <instanceOf xlink:href="#tDevice" />
          <description>
            A device that is capable of cellular mobile communication like GSM, GPRS or UMTS
          </description>
        </topic>
        <topic id="rGpsDevice">
          <instanceOf xlink:href="#tDevice" />
          <description>
            A device that is equipped with a GPS receiver, therefore able to locate itself.
          </description>
        </topic>
        <topic id="rMobileContract">
          <instanceOf xlink:href="#tMobileContract" />
          <description>
            A subscription of the user for mobile services, used for payment information.
          </description>
        </topic>
        <topic id="rMobileProvider">
          <instanceOf xlink:href="#tMobileProvider" />
          <description>
            A company offering mobile services.
          </description>
        </topic>
        <topic id="rLocationMobile">
          <description>
            A temporary topic representing the location of the user determined by
            a mobile provider.
          </description>
          <attribute>
            <baseName>
              <baseNameString>long</baseNameString>
            </baseName>
          </attribute>
          <attribute>
            <baseName>
              <baseNameString>lat</baseNameString>
            </baseName>
          </attribute>
        </topic>
        <topic id="rLocationGps">
          <description>

```

```

    A temporary topic representing the location of the user determined by
    a gps device.
  </description>
  <attribute>
    <baseName>
      <baseNameString>long</baseNameString>
    </baseName>
  </attribute>
  <attribute>
    <baseName>
      <baseNameString>lat</baseNameString>
    </baseName>
  </attribute>
</topic>

<activeAssociation id="user_has_mobileDevice">
  <description>
    Determines all devices of the user that are able for mobile communication.
  </description>
  <member><topicRef xlink:href="#rUser" /></member>
  <member><topicRef xlink:href="#rMobileDevice" /></member>
  <codeBlock>
    <![CDATA[
      void user_has_mobileDevice (Topic rUser, Vector rMobileDevice) {
        // get all devices
        Vector allDevices = rUser.followAssoc("person_uses_device");

        // select the devices with mobile capabilities
        Iterator iter = allDevices.iterator();
        while (iter.hasNext()) {
          Topic device = (Topic)iter.next();
          if device.isMobile() rMobileDevice.add(device);
        }
      }
    ]]>
  </codeBlock>
</activeAssociation>

<activeAssociation id="mobileDevice_has_mobileContract">
  <description>
    Determines the mobile contract that is used with this mobile device.
  </description>
  <member><topicRef xlink:href="#rMobileDevice" /></member>
  <member><topicRef xlink:href="#rMobileContract" /></member>
  <codeBlock>
    <![CDATA[
      void mobileDevice_has_mobileContract (Topic rMobileDevice,
        Vector rMobileContract) {
        rMobileContract.add( rMobileDevice.followAssoc("device_has_contract"));
      }
    ]]>
  </codeBlock>
</activeAssociation>

<activeAssociation id="mobileContract_with_mobileProvider">
  <description>
    Determines the mobile provider, with which the given
    mobile contract is made.
  </description>
  <member><topicRef xlink:href="#rMobileContract" /></member>
  <member><topicRef xlink:href="#rMobileProvider" /></member>
  <codeBlock>
    <![CDATA[
      void mobileContract_with_mobileProvider (Topic rMobileContract,
        Vector rMobileProvider) {
        rMobileProvider.add(rMobileContract.followAssoc("contract_with_company"));
      }
    ]]>
  </codeBlock>
</activeAssociation>

<activeAssociation id="mobileProvider_locates_mobileDevice">

```

```

<description>
  Calls the mobile provider to locate the given device.
  Payment is done using the given contract.
</description>
<member><topicRef xlink:href="#rMobileProvider" /></member>
<member><topicRef xlink:href="#rMobileDevice" /></member>
<member><topicRef xlink:href="#rMobileContract" /></member>
<member><topicRef xlink:href="#rLocationMobile" /></member>
<codeBlock>
  <![CDATA[
    void mobileProvider_locates_mobileDevice
      (Topic rMobileProvider, Topic rMobileDevice, Topic rMobileContract,
       Vector rLocationMobile) {
      Vector providerLocation = rMobileProvider.getDeviceCoordinates(
        rMobileDevice.getID(), rMobileContract.getID());
      Topic loc = cloneTopic("#rLocationMobile");
      loc.setAttribute(long, providerLocation.get(0));
      loc.setAttribute(lat, providerLocation.get(1));
      rLocationMobile.add(loc);
    }
  ]]>
</codeBlock>
</activeAssociation>

<activeAssociation id="user_has_gpsDevice">
  <description>
    Determines all devices of the user that are able to use GPS.
  </description>
  <member><topicRef xlink:href="#rUser" /></member>
  <member><topicRef xlink:href="#rGpsDevice" /></member>
  <codeBlock>
    <![CDATA[
      void user_has_gpsDevice (Topic rUser, Vector rGpsDevice) {
        // get all devices
        Vector allDevices = rUser.followAssoc("person_uses_device");

        // select the devices with GPS capabilities
        Iterator iter = allDevices.iterator();
        while (iter.hasNext()) {
          Topic device = (Topic)iter.next();
          if device.hasGPS() rGpsDevice.add(device);
        }
      }
    ]]>
  </codeBlock>
</activeAssociation>

<activeAssociation id="gpsDevice_determines_location">
  <description>
    Calls the GPS on the devices to get its current location.
  </description>
  <member><topicRef xlink:href="#rGpsDevice" /></member>
  <member><topicRef xlink:href="#rLocationGps" /></member>
  <codeBlock>
    <![CDATA[
      void gpsDevice_determines_location (Topic rGpsDevice, Vector rLocation) {
        Vector gpsLocation = rGpsDevice.getCurrentGpsLocation();
        Topic loc = cloneTopic("#rLocationGps");
        loc.setAttribute(long, gpsLocation.get(0));
        loc.setAttribute(lat, gpsLocation.get(1));
        rLocationGps.add(loc);
      }
    ]]>
  </codeBlock>
</activeAssociation>

<activeAssociation id="location_fusion">
  <description>
    Determines the current location of the user by performing a
    fusion operation on the given locations.
    - use GPS location if available
    - otherwise use mobile location
  </description>

```

```

</description>
<member><topicRef xlink:href="#rLocationMobile" /></member>
<member><topicRef xlink:href="#rLocationGPS" /></member>
<member><topicRef xlink:href="#user" /></member>
<codeBlock>
  <![CDATA[
    void location_fusion (Topic rMobileLocation, Topic rGpsLocation, Topic user) {
      if (rGpsLocation.getAttribute("long") <> null) {
        user.setAttribute("location",
          ConvertLatLong2WGS84(rGpsLocation.getAttribute("long"),
            rGpsLocation.getAttribute("lat")));
      } else {
        user.setAttribute("location",
          ConvertLatLong2WGS84(rMobileLocation.getAttribute("long"),
            rMobileLocation.getAttribute("lat")));
      }
    }
  ]]>
</codeBlock>
</activeAssociation>

</map>
</accessMap>
</attribute>
</topic>

<topic id="rMobileContract">
  <baseName>
    <baseNameString>Mobile Contract</baseNameString>
  </baseName>
  <description>A contract for mobile services</description>
  <instanceOf>
    <topicRef xlink:href="#cMobileContract" />
  </instanceOf>
</topic>

<topic id="rMobileProvider">
  <baseName>
    <baseNameString>Mobile Provider</baseNameString>
  </baseName>
  <description>A mobile Provider of the user</description>
  <instanceOf>
    <topicRef xlink:href="#cMobileProvider" />
  </instanceOf>
</topic>

<topic id="rVoiceProfile">
  <baseName>
    <baseNameString>Voice Profile</baseNameString>
  </baseName>
  <description>the voice profile of the user</description>
  <instanceOf>
    <topicRef xlink:href="#cVoiceProfile" />
  </instanceOf>
</topic>

<topic id="rFoodPreferences">
  <baseName>
    <baseNameString>Food Preferences</baseNameString>
  </baseName>
  <description>the food preferences of the user</description>
  <instanceOf>
    <topicRef xlink:href="#cFoodPreferences" />
  </instanceOf>
</topic>

<!-- Nutzer hat Mobilfunkvertrag mit Mobilfunkanbieter -->
<association id="user_has_mobileContract_with_mobileProvider">
  <member><topicRef xlink:href="#rUser"/></member>
  <member><topicRef xlink:href="#rMobileContract"/></member>

```

```

    <member><topicRef xlink:href="#rMobileProvider"/></member>
</association>

<!-- Nutzer hat Stimmprofil -->
<association id="user_has_voiceProfile">
    <member><topicRef xlink:href="#rUser"/></member>
    <member><topicRef xlink:href="#rVoiceProfile"/></member>
</association>

<!-- Nutzer hat Speisepräferenzen -->
<association id="user_has_foodPreferences">
    <member><topicRef xlink:href="#rUser"/></member>
    <member><topicRef xlink:href="#rFoodPreferences"/></member>
</association>

<!-- Nutzer hat Terminplaner -->
<association id="user_has_diary">
    <member><topicRef xlink:href="#rUser"/></member>
    <member><topicRef xlink:href="#rDiary"/></member>
</association>

<!-- Nutzer hat Messenger -->
<association id="user_has_messenger">
    <member><topicRef xlink:href="#rUser"/></member>
    <member><topicRef xlink:href="#rMessenger"/></member>
</association>

<!-- Nutzer hat Adressbuch -->
<association id="user_has_addressBook">
    <member><topicRef xlink:href="#rUser"/></member>
    <member><topicRef xlink:href="#rAddressBook"/></member>
</association>

<!-- Terminplaner enthält Termine -->
<activeAssociation id="diary_contains_appointments">
    <description>
        Represents all appointments in the given diary as topics in the map.
    </description>
    <member><topicRef xlink:href="#rDiary"/></member>
    <member><topicRef xlink:href="#rAppointment"/></member>
    <codeBlock>
        <![CDATA[
            void diary_contains_appointment (Topic diary, Vector appointments) {
                // call the access operation "getAllAppointments()"
                // of the type "diary" to get all entries in the diary and return them.
                appointments = diary.getAllAppointments();
            }
        ]]>
    </codeBlock>
</association>

<!-- Messenger enthält Nachrichten -->
<activeAssociation id="messenger_contains_messages">
    <description>
        Represents all messages in the given messenger as topics in the map.
    </description>
    <member><topicRef xlink:href="#rMessenger"/></member>
    <member><topicRef xlink:href="#rMessage"/></member>
    <codeBlock>
        <![CDATA[
            void messenger_contains_messages (Topic messenger, Vector messages) {
                // call the access operation "getAllMessages()"
                // of the type "messenger" to get all messages in the diary
                messages = diary.getAllAppointments();
            }
        ]]>
    </codeBlock>
</association>

<!-- Adressbuch enthält Kontakte -->
<activeAssociation id="addressbook_contains_aquaintances">
    <description>

```



```

    Represents all contacts in the given address book as topics in the map.
  </description>
  <member><topicRef xlink:href="#rAddressbook" /></member>
  <member><topicRef xlink:href="#rAquaintance" /></member>
  <codeBlock>
    <![CDATA[
      void addressbook_contains_aquaintances (Topic addressbook, Vector acquaintance) {
        // call the access operation "getAllContacts()"
        // of the type "addressbook" to get all acquaintances in the addressbook
        acquaintances = diary.getAllContacts();
      }
    ]]>
  </codeBlock>
</association>

</map>
</sceneMaps>
</realworldUniverse>

```

Anhang E.2.3 Kontextquellentypenebene

An dieser Stelle werden die vom Nutzerprofil verwendeten Kontextquellen samt ihrer Zugriffsoperationen definiert.

```

<electronicUniverse>

  <contextSource>
    <realworldObjectType><topicRef xlink:href="#tMobileProvider" /></realworldObjectType>
    <uriSchema>mobileProvider</uriSchema>
    <accessOperation>
      <name>getDeviceCoordinates</name>
      <description>
        Returns the coordinates of the specified device.
        Payment is done using the information in the given contract.
      </description>
      <parameter>
        <name>deviceID</name>
        <type>String</type>
      </parameter>
      <parameter>
        <name>contract-ID</name>
        <type>String</type>
      </parameter>
      <result>
        <type>http://www.cm.org/standards/types.cm#longLatLocation</type>
      </result>
    </accessOperation>
  </contextSource>

  <contextSource>
    <realworldObjectType><topicRef xlink:href="#tDevice" /></realworldObjectType>
    <uriSchema>device</uriSchema>
    <accessOperation>
      <name>getDeviceCoordinates</name>
      <description>
        Returns the coordinates of the specified device.
        Payment is done using the information in the given contract.
      </description>
      <parameter>
        <name>deviceID</name>
        <type>String</type>
      </parameter>
      <parameter>
        <name>contract-ID</name>
        <type>String</type>
      </parameter>
      <result>
        <type>http://www.cm.org/standards/types.cm#longLatLocation</type>
      </result>
    </accessOperation>
  </contextSource>

```

```

    </accessOperation>
  </contextSource>

  Device
  - isMobile
  - hasGPS

  gpsDevice
  - getCurrentGpsLocation

<electronicUniverse>

```

Anhang E.2.4 Realweltmodellinstanz

An dieser Stelle wird das in den Anwendungsbeispielen verwendete, auf dem eben definierten Realweltuniversum basierende Nutzerprofil vorgestellt, natürlich in Form einer *Situation*. Dieses ist unter der URL <http://www.jeffmalone.de/userprofile.cm> abrufbar.

```

<contextMap>
  <realworldUniverseRef xlink:href="http://www.cm.org/standards/user.profile.cm" />
  <instances>
    <map id="UserProfile_JeffMalone"
      instanceOf xlink:href="#smUserProfile">

      <!-- Avatar der Person "John Doe" -->
      <topic id="iPerson_123_john_doe">
        <instanceOf>
          <topicRef xlink:href="http://www.cm.org/standards/user.profile.cm#Human" />
        </instanceOf>
        <baseName>
          <baseNameString>John Doe</baseNameString>
          <variant>
            <parameters>
              <subjectIndicatorRef
                xlink:href="http://www.cm.org/standards/basenameVariantTypes.cm#nickname">
              </parameters>
              <resourceData>Jodo</resourceData>
            </variant>
          </baseName>
          <attribute>
            <basename>
              <baseNameString>First Name</baseNameString>
            </basename>
            <attributeValue>
              <standardizedValue>
                <value>John</value>
              </standardizedValue>
            </attributeValue>
            <quality>
              <qualityDescriptor>
                <qualityParameter
                  topicRef="http://www.cm.org/standards/qualityTypes.cm#certainty" />
                <qualityMetrics>
                  <standardizedValue>
                    <unit topicRef="http://www.cm.org/standards/metrics.cm#percent" />
                    <value>100</value>
                  </standardizedValue>
                </qualityMetrics>
              </qualityDescriptor>
            </quality>
            <security>
              <securityDescriptor type="accessible">
                <actor><topicRef="http://www.cm.org/standards/actors.cm#all" /></actor>
              </securityDescriptor>
              <securityDescriptor type="copyable">
                <actor><topicRef="http://www.cm.org/standards/actors.cm#all" /></actor>
              </securityDescriptor>
              <securityDescriptor type="fusible">

```

```
<actor><topicRef="http://www.cm.org/standards/actors.cm#none" /></actor>
</securityDescriptor>
</security>
</attribute>
<attribute>
  <basename>
    <baseNameString>Last Name</baseNameString>
  </basename>
  <attributeValue>
    <standardizedValue>
      <value>Doe</value>
    </standardizedValue>
  </attributeValue>
  <quality>
    <qualityDescriptor>
      <qualityParameter>
        topicRef="http://www.cm.org/standards/qualityTypes.cm#certainty" />
      <qualityMetrics>
        <standardizedValue>
          <unit topicRef="http://www.cm.org/standards/metrics.cm#percent" />
          <value>100</value>
        </standardizedValue>
      </qualityMetrics>
    </qualityDescriptor>
  </quality>
</security>
<security>
  <securityDescriptor type="accessible">
    <actor><topicRef="http://www.cm.org/standards/actors.cm#all" /></actor>
  </securityDescriptor>
  <securityDescriptor type="copyable">
    <actor><topicRef="http://www.cm.org/standards/actors.cm#all" /></actor>
  </securityDescriptor>
  <securityDescriptor type="fusible">
    <actor><topicRef="http://www.cm.org/standards/actors.cm#none" /></actor>
  </securityDescriptor>
</security>
</attribute>
<attribute>
  <basename>
    <baseNameString>location</baseNameString>
  </basename>
  <attributeValue>
    <standardizedValue>
      <unit topicRef="http://www.cm.org/standards/metrics.cm#WGS84" />
      <value>N52° 31.0' E13° 22.27'</value>
    </standardizedValue>
  </attributeValue>
  <history>
    <historicalValue timeStamp="2005-03-27--14:13:57"
      value="N52°31.1' E13°22.57'" />
    <historicalValue timeStamp="2005-03-27--08:25:11"
      value="N52°31.2' E13°25.45'" />
  </history>
  <quality>
    <qualityDescriptor>
      <qualityParameter>
        topicRef="http://www.cm.org/standards/qualityTypes.cm#accuracy" />
      <qualityMetrics>
        <standardizedValue>
          <unit topicRef="http://www.cm.org/standards/metrics.cm#meters" />
          <value>50</value>
        </standardizedValue>
      </qualityMetrics>
    </qualityDescriptor>
  </quality>
</security>
<security>
  <securityDescriptor type="accessible">
    <actor><topicRef="http://www.cm.org/applications/scanav.cm#scanav"/></actor>
  </securityDescriptor>
</security>
<ynamics>
  <dynamicsDescriptor>
    <minTimeBetweenUpdates>
```

```

        <standardizedValue>
          <unit topicRef="http://www.cm.org/standards/metrics.cm#second" />
          <value>10</value>
        </standardizedValue>
      </minTimeBetweenUpdates>
      <maxTimeBetweenUpdates>
        <standardizedValue>
          <unit topicRef="http://www.cm.org/standards/metrics.cm#hour" />
          <value>24</value>
        </standardizedValue>
      </maxTimeBetweenUpdates>
    </dynamicsDescriptor>
  </dynamics>
</attribute>
<attribute>
  <basename>
    <baseNameString>activity</baseNameString>
  </basename>
  <description>
    The current activity of the user.
    This attribute is set manually by assistants of the user.
  </description>
  <attributeValue>
    <topicRef xlink:href="http://www.cm.org/standards/activities.cm#meeting" />
  </attributeValue>
  <history>
    <historicalValue timeStamp="2005-03-26--22:53:28"
      value="http://www.cm.org/standards/activities.cm#sleeping" />
    <historicalValue timeStamp="2005-03-27--11:25:42"
      value="http://www.cm.org/standards/activities.cm#walking" />
    <historicalValue timeStamp="2005-03-27--08:37:36"
      value="http://www.cm.org/standards/metrics.cm#driving_car" />
  </history>
  <quality>
    <qualityDescriptor>
      <qualityParameter
        topicRef="http://www.cm.org/standards/qualityTypes.cm#certainty" />
      <qualityMetrics>
        <standardizedValue>
          <unit topicRef="http://www.cm.org/standards/metrics.cm#percent" />
          <value>80</value>
        </standardizedValue>
      </qualityMetrics>
    </qualityDescriptor>
  </quality>
  <security>
    <securityDescriptor type="accessible">
      <actor><topicRef="http://www.cm.org/applications/scanav.cm#iScanav"/></actor>
    </securityDescriptor>
  </security>
  <dynamics>
    <dynamicsDescriptor>
      <minTimeBetweenUpdates>
        <standardizedValue>
          <unit topicRef="http://www.cm.org/standards/metrics.cm#second" />
          <value>10</value>
        </standardizedValue>
      </minTimeBetweenUpdates>
      <maxTimeBetweenUpdates>
        <standardizedValue>
          <unit topicRef="http://www.cm.org/standards/metrics.cm#hour" />
          <value>24</value>
        </standardizedValue>
      </maxTimeBetweenUpdates>
    </dynamicsDescriptor>
  </dynamics>
</attribute>
<attribute>
  <baseName>
    <baseNameString>Accessibility</baseNameString>
  </baseName>
  <description>
    Indicates if the user wants to be disturbed, e.g. by incoming phone calls.
  </description>

```

```
</description>
<accessMap>
  <map>
    <topic id="user"
      xlink:href="#iPerson_123_john_doe" />
    <activeAssociation id="accessibility_by_activity">
      <member>
        <topicRef xlink:href="#user"
        </member>
      <codeBlock>
        <![CDATA[
          Topic tUser = getTopicByID("#user");
          String activity = tUser.getAttributeByName("activity");
          String accessibility = "yes";
          if (activity == "http://www.cm.org/standards/activities.cm#sleeping")
            accessibility = "http://www.cm.org/standards/boolean.cm#no";
          if (activity == "http://www.cm.org/standards/activities.cm#meeting")
            accessibility = "http://www.cm.org/standards/boolean.cm#no";
          tUser.setAttribute("accessibility", accessibility);
        ]>
      </codeBlock>
    </activeAssociation>
  </map>
</accessMap>
</attribute>
</topic>
</map>
</instances>
</contextMap>
```

Ehrenwörtliche Erklärung

Hiermit versichere ich, die vorliegende Arbeit selbständig, ohne fremde Hilfe und ohne Benutzung anderer als der von mir angegebenen Quellen angefertigt zu haben. Alle aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche gekennzeichnet. Die Arbeit wurde noch keiner Prüfungsbehörde in gleicher oder ähnlicher Form vorgelegt.

Dresden, den 7. Februar 2007