



**EffizienzCluster
LogistikRuhr**

Verbundprojekt OrGoLo:

Organisatorische Innovationen mit Good Governance in Logistik-Netzwerken

Dipl.-Inf. Martin Kowalski, Dipl.-Wirt.-Inf. Dennis Kater

Case-based Reasoning in Supply Chains

– Qualitatives Case Retrieval –

Förderkennzeichen: 01IC10L20A



GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung

OrGoLo-Projektbericht Nr. 9

ISSN 1866-9255

Abstract

Um nachhaltige Wettbewerbsvorteile erzielen zu können und in langfristige betriebswirtschaftliche Erfolge umzusetzen, ist ein „ganzheitliches“ Verständnis der Struktur von und der Geschäftsprozesse in internationalen Supply Chains erforderlich. Dafür ist eine Neuausrichtung des Gestaltungsverständnisses für Supply Chains erforderlich, das die bislang dominierende Orientierung an rein quantitativen und „harten“ Erfolgskriterien für das operative Supply Chain Management keineswegs vernachlässigt. Aber diese neue Sichtweise wird um zusätzliche qualitative und „weiche“ Einflussgrößen des vor allem strategischen Supply Chain Managements erweitert, die sich oftmals nur indirekt, aber umso nachhaltiger auf Wettbewerbsfähigkeit und Unternehmenserfolg auszuwirken vermögen. Die qualitativen und „weichen“ Einflussgrößen lassen sich nicht mehr mit simplen Kennzahlen und entsprechenden numerischen „Daten“ über Geschäftsprozesse adäquat erfassen, sondern erfordern komplexe kognitive Strukturen, die im Allgemeinen als „Wissen“ bezeichnet werden. Es sind erste Ansätze vorhanden, die es ermöglichen, z.B. erfahrungsbasiertes Wissen zu verarbeiten. Im Zentrum dieser Wissensverarbeitungstechniken stehen die Technik des Case-based Reasonings (CBR), die aus der Erforschung Künstlicher Intelligenz (KI) stammt, und die Technik der Ontologien. Ontologien dienen vor allem dazu, natürlichsprachlich repräsentiertes Wissen so aufzubereiten, dass es von Computern – gemeint ist hiermit vor allem entsprechende Software – inhaltlich verstanden und somit auch auf der semantischen Ebene, d.h. „verständnisvoll“ verarbeitet werden kann. In dieser Arbeit wird gezeigt, wie die Techniken des Case-based Reasonings und der Ontologien in einen konkreten und anspruchsvollen betriebswirtschaftlichen Anwendungskontext – das Management von Supply-Chain-Projekten – eingesetzt werden können. Des Weiteren wird anhand von konkreten Fällen aus dem Bereich internationaler Supply-Chain-Projekte ein qualitatives Case Retrieval durchgeführt.

Inhaltsverzeichnis

	<u>Seite</u>
Abkürzungs-, Akronym- und Symbolverzeichnis	V
Abbildungsverzeichnis.....	X
Tabellenverzeichnis.....	XI
1 Motivation.....	1
1.1 Supply Chain Governance	1
1.2 Wissenschaftliches Problem	1
1.3 Aufbau der Arbeit	3
2 Case-based Reasoning	4
2.1 Die Intention des Case-based Reasonings	4
2.2 Ein Fall im Case-based Reasoning.....	5
2.3 Wissensrepräsentation in der Fallbasis	5
2.4 Anwendungsformen des Case-based Reasonings	7
2.4.1 Strukturelles Case-based Reasoning.....	7
2.4.2 Textbasiertes Case-based Reasoning.....	8
2.4.3 Dialogorientiertes Case-based Reasoning	9
2.5 Case-based-Reasoning-Zyklus.....	9
2.5.1 Einleitung.....	9
2.5.2 Retrieve.....	10
2.5.3 Reuse.....	12
2.5.4 Revise	13
2.5.5 Retain.....	13
3 Ontologien.....	13
3.1 Einführung in das Themenfeld „Ontologien“	13
3.1.1 Einleitung.....	13
3.1.2 Philosophischer Ursprung.....	14
3.1.3 Modernes Ontologieverständnis	15
3.1.4 Konzeptualisierung	16
3.1.5 Spezifikation	17
3.2 Repäsentationsformen von Ontologien.....	19

3.2.1	Logik.....	19
3.2.2	Semantische Netze.....	20
4	Ein Prototyp für ontologiestütztes Case-based Reasoning in Supply Chains	21
4.1	Definition der Vorgehensweise für die Entwicklung eines ontologiestützten Case-based-Reasoning Systems.....	21
4.2	Entwicklung des Prototyps.....	24
4.2.1	Konzept.....	24
4.2.1.1	Anforderungsanalyse	24
4.2.1.2	Systemanalyse	24
4.2.2	Entwurf	26
4.2.2.1	Auswahl der Softwarekomponenten.....	26
4.2.2.2	Protégé.....	27
4.2.2.3	MyCBR.....	27
4.2.2.4	Integration der beiden Komponenten	28
4.2.3	Implementierung.....	29
4.2.3.1	Umfang der Implementierung	29
4.2.3.2	Erstellung der Ontologie.....	29
4.2.3.2.1	Vorgehen zur Ontologierstellung.....	29
4.2.3.2.2	Beschreibung der Ontologierstellung	30
4.2.3.2.3	Wiederverwendung bestehender Ontologien.....	31
4.2.3.2.4	Identifikation relevanter Begriffe	31
4.2.3.2.5	Festlegung der Klassen und der Klassenhierarchie	32
4.2.3.2.6	Definition der Slots (Attribute und Relationen)	34
4.2.3.2.7	Definition der Slot-Eigenschaften	35
4.2.3.3	Erstellung von Ähnlichkeitsmaßstäben	44
4.2.3.3.1	Lokale Ähnlichkeitsmaßstäbe.....	44
4.2.3.3.2	Globale Ähnlichkeitsmaßstäbe	52
4.2.3.4	Erstellung des Retrieval-Algorithmus	55
4.2.4	Test	56
4.2.4.1	Anlegen einer Fallbasis mit Testfällen	56
4.2.4.2	Test des Retrievals für das Case-based Reasoning.....	56
5	Fazit und Ausblick.....	59
6	Literaturverzeichnis	61

7	Anhang 1: Falldatenbank.....	69
8	Anhang 2: Testvergleich zwischen Fall 1 und Fall 2	87

Abkürzungs-, Akronym- und Symbolverzeichnis

AG	Aktiengesellschaft
AHP	Analytic Hierarchy Process
API	Application Programming Interface
ASEAN	Association of Southeast Asian Nations
Aufl.	Auflage
BMIR	Biomedical Informatics Research
bzw.	beziehungsweise
CACM	Central American Common Market
CAN	Comunidad Andina de Naciones
CARICOM	Caribbean Community and Common Market
CBR	Case-Based Reasoning
CCBR	Conversational Case-Based Reasoning
CEMAC	Communauté Économique et Monétaire de l'Afrique Centrale
CFR	Cost And Freight
CIF	Cost Insurance Freight
CIP	Carriage Insurance Paid
COMESA	Common Market for Eastern and Southern Africa
CPT	Carriage Paid To
CSR	Corporate Social Responsibility
CSV	Character Separated Values (auch: Comma Separated Values)
CYRUS	Computerized Yale Retrieval and Update System

DAP	Delivered At Place
DAT	Delivered At Terminal
DDP	Delivered Duty Paid
d.h.	das heißt
dt.	deutsch
EAC	East African Community
ECOWAS	Economic Community of West African States
eds.	Editors
engl.	englisch
et al.	et alii
etc.	et cetera
EU	Europäische Union
EurAsEC	Eurasian Economic Community
EXW	Ex Works
f.	folgende
FAQ	Frequently Asked Questions
FAS	Free Alongside Ship
FCA	Free Carrier
ff.	fort folgende
FOB	Free On Board
GB	Gigabyte
GCC	Cooperation Council for the Arab States of the Gulf

ggf.	gegebenenfalls
GHz	Gigahertz
GmbH	Gesellschaft mit beschränkter Haftung
GPL	General Public License
GUI	Graphical User Interface
Hrsg.	Herausgeber
ID3	Iterative Dichotomiser 3
i.d.R.	in der Regel
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
kg	Kilogramm
KI	Künstliche Intelligenz
LKW	Lastkraftwagen
Mio.	Millionen
Mercosur	Mercado Común del Sur
MOP	Memory Organisation Packets
NAFTA	North American Free Trade Agreement
No.	Number
Nr.	Nummer
OKBC	Open Knowledge Base Connectivity
OrGoLo	Organisatorische Innovationen mit Good Governance in Logistik-Netzwerken
o. S.	ohne Seitenangabe

o.V.	ohne Verfasser
OWL	Web Ontology Language
PIF	Pacific Islands Forum
RAM	Random Access Memory
RDF	Resource Description Framework
S.	Seite
SAARC	South Asian Association for Regional Cooperation
SACU	Southern African Customs Union
SCBR	Strukturiertes Case-Based Reasoning
SCG	Supply Chain Governance
s.o.	siehe oben
SQL	Structured Query Language
TARIC	Tarif intégré des Communautés européennes
TCBR	Textbasiertes Case-Based Reasoning
u.a.	unter anderem
UK	United Kingdom
UN	United Nations
UNECE	United Nations Economic Commission for Europe
URL	Uniform Resource Locator
USA	United States of America
vgl.	vergleiche
Vol.	Volume

XML Extensible Markup Language

z.B. zum Beispiel

Abbildungsverzeichnis

Abbildung 1: Aufbau der Arbeit	3
Abbildung 2: Memory Organization Packets	6
Abbildung 3: Ausschnitt aus dem Dynamic Memory Modell von CYRUS.....	7
Abbildung 4: Werte des relationalen Attributs "Speichermedium"	8
Abbildung 5: CBR-Zyklus	10
Abbildung 6: Entscheidungsbaum beim Induktiven Retrieval	12
Abbildung 7: Semiotisches Dreieck.....	18
Abbildung 8: Beispiel für ein semantisches Netz	21
Abbildung 9: Klassisches Wasserfallmodell.....	22
Abbildung 10: Vorgehensweise zur Entwicklung des ontologiegestützten CBR-Prototyps	23
Abbildung 11: Entity-Relationship-Modell	25
Abbildung 12: In Protégé dargestellte Klassenhierarchie	33
Abbildung 13: Relationen innerhalb der Ontologie	39
Abbildung 14: Phyton-Skript für relative Distanz	46
Abbildung 15: Phyton-Skript für Postleitzahlen	47
Abbildung 16: Taxonomie für Lieferbedingungen	50
Abbildung 17: Auszug aus der Taxonomie für Weltregionen	50
Abbildung 18: Protégé-Screenshot: Auszug aus Warenart-Taxonomie	51
Abbildung 19: Skript für Ähnlichkeitsbestimmung von TARIC-Nummern	51
Abbildung 20: Ähnlichkeitswert für die Subklassen zur Klasse Akteur	53

Tabellenverzeichnis

Tabelle 1: Relevante Begriffe für die Ontologierstellung	32
Tabelle 2: Übersicht der verwendeten Slots (Teil 1).....	37
Tabelle 3: Übersicht der verwendeten Slots (Teil 2).....	38
Tabelle 4: Gefahrgutklassen.....	41
Tabelle 5: Handelszonen	42
Tabelle 6: Ausprägungen von Einflussfaktoren (Prioritäten)	42
Tabelle 7: Incoterms 2010.....	43
Tabelle 8: Ähnlichkeitstabelle Prioritäten.....	48
Tabelle 9: Ähnlichkeitstabelle Unternehmensbeziehung.....	48
Tabelle 10: Ähnlichkeitstabelle Unternehmensgröße	49
Tabelle 11: Gewichtungen der Attribute und Relationen (Teil 1)	54
Tabelle 12: Gewichtungen des Retrieval -Algorithmus (Teil 2).....	55
Tabelle 13: Retrievalergebnisse	57
Tabelle 14: Testergebnis	58

1 Motivation¹

1.1 Supply Chain Governance

Im Zuge der anhaltenden Globalisierung der Weltwirtschaft tauchen in öffentlichen Diskussionen und wissenschaftlichen Publikationen vermehrt Schlagwörter wie Nachhaltigkeit², Good Governance³ und Corporate Social Responsibility (CSR)⁴ auf, die für ein Umdenken in Politik und Wirtschaft stehen. In Bezug auf die Betriebswirtschaft bedeuten aktuelle Entwicklungen, dass sich zu den konventionellen und quantitativ messbaren ökonomischen Einflussfaktoren wie Kosten und Erlösen ökologisch, rechtlich und gesellschaftlich relevante Aspekte gesellen. Diese Komplexitätssteigerung (in Bezug auf die Erweiterung der in Entscheidungsprozessen zu beachtenden Einflussfaktoren) überträgt sich in der Logistik auf das Supply Chain Management.⁵

Die Wissenschaft und die betriebliche Praxis teilen zunehmend die Ansicht, dass ein ganzheitliches Strukturverständnis der Geschäftsprozesse in Supply Chains erforderlich ist, um nachhaltige Wettbewerbsvorteile zu erzielen. Zusätzlich werden in unternehmensübergreifenden Wertschöpfungsnetzwerken häufig vermeidbare Effizienzverluste beobachtet, die darauf zurückzuführen sind, dass eine gemeinsame Sicht der Beteiligten auf ein gesamtes Wertschöpfungsnetzwerk fehlt – die Prozessoptimierung der einzelnen Akteure steht bislang noch im Vordergrund, ohne besondere Rücksicht auf die Gesamtwirkung der Handlungen einzelner Akteure.⁶

Aus dieser Beobachtung geht das Supply Chain Governance als neuartige Leitidee⁷ hervor. Supply Chain Governance bereichert das konventionelle Supply Chain Management um Nachhaltigkeitsaspekte⁸ sowie rechtliche, ökologische und gesellschaftliche Anforderungen und steigert damit auch die organisatorische Komplexität in der Gestaltung von Lieferketten.⁹

1.2 Wissenschaftliches Problem

Die Erweiterung des Supply Chain Managements um die Aspekte des Supply Chain Governance führt zu einer neuartigen Problemstruktur, deren Komplexität es zu bewältigen gilt. Es ist betriebswirtschaftlich wünschenswert, die nachhaltige Gestaltung komplexer, internationaler Lieferketten durch intelligente Software zu unterstützen, um die neuen Herausforderungen ökonomisch effizient bewältigen zu können. Für die Entwicklung einer wissensbasierten und lernfähigen Software, die

1) Diese Arbeit stellt eine Überarbeitung von KATER (2011) dar.

2) Vgl. ALLEN (2008), S. 187 ff.; DALY (2007), S. 36 ff.; BARBIER/PEARCE (2000), S. 1 ff.

3) Vgl. NUSCHELER (2009), S. 10 ff.; STÜTZER (2007), S. 10 ff.

4) Vgl. MALLIN (2009), S. 1 ff.; CURBACH (2009), S. 1 ff.; BECKMANN (2007), S. 31 ff.

5) Vgl. PETERS (2010), S. 1 f.; FERGER (2009), S. 27 ff.; WILHELM (2009), S. 133 ff.; BUTCHER/LALWANI/MANGAN (2008), S. 267 ff.; MÜLLER/SEURING (2008), S. 455 ff.; ROOME (1998), S. 88 f.

6) Vgl. ZELEWSKI (2011), S. 7.

7) Vgl. ZELEWSKI (2011), S. 2.

8) Vgl. BITRAN/GURUMURTHIO/SAM (2006), S. 1 ff.

9) Die Begriffe „Supply Chain“ und „Lieferkette“ werden hier synonym verwendet.

einen solchen Mehrwert leisten könnte, bietet sich eine Methode¹⁰ aus der Künstlichen Intelligenz an: das Case-Based Reasoning (CBR).¹¹

Der Einsatz des Case-Based Reasonings empfiehlt sich besonders dann, wenn Entscheidungsprobleme eines Anwendungsumfelds zu komplex¹² sind, um sie sicher anhand von Regeln oder Algorithmen lösen zu können¹³. Die vielen Einflussfaktoren und komplizierten interdisziplinären Zusammenhänge machen Supply Chain Management somit zu einem idealen Einsatzgebiet für das Case-based Reasoning.

Bislang wurde Case-Based Reasoning hauptsächlich in Bereichen eingesetzt, in denen sich die Beschreibung des Problemumfelds auf einen kleinen Ausschnitt der Realwelt beschränkt und die Ausprägungen von Problemfällen hauptsächlich quantitativ beschreibbar sind, was eine Ähnlichkeitsmessung im Case-Based Reasoning durch Techniken wie beispielsweise *Nearest-Neighbour-Verfahren*¹⁴ ermöglicht. Erfolgreiche CBR-Anwendungen finden sich unter Anderem in Einsatzgebieten wie dem technischen Konstruktionswesen¹⁵, der technischer Diagnose¹⁶, der medizinischen Diagnose¹⁷, dem Customer Support¹⁸ und im E-Commerce.¹⁹

Besonderer Anspruch im Anwendungsgebiet Supply Chain Management ist die, im Gegensatz zu herkömmlichen Einsatzgebieten von CBR-Systemen, vielfältigere und kompliziertere Struktur von Problemfällen. Ein Case-Based Reasoning-System für SCM muss Problemstellungen einer äußerst komplexen Domäne mit vielen verschiedenen Einflussfaktoren (Zoll, Umweltfreundlichkeit²⁰, öffentliches Interesse an Nachhaltigkeit etc.) bewältigen. Die Einflussfaktoren sind zudem meist schwer mittels Kennzahlen oder mathematischer Größen quantitativ auszudrücken, sondern erfordern eine qualitative Untersuchung.

-
- 10) Case-Based Reasoning wird hier als *Methode* verstanden, die verschiedene *Techniken* verwendet. Vgl. WATSON (1999), S. 303 ff.
 - 11) Vgl. AAMODT/PLAZA (1994), S. 39 ff.; KOLODNER (1992), S. 1 ff.
 - 12) Komplexität hinsichtlich nicht prognostizierbarer Ausprägungen von Einflussfaktoren auf eine Problemdomäne, wie z. B. menschliches Verhalten.
 - 13) Vgl. KOLODNER (1992), S. 29.
 - 14) Vgl. OKAMOTO/SATOH (1994), S. 101 ff.
 - 15) Beispielsweise wird das Programm CLAVIER zur Unterstützung der Ingenieure beim Layout von speziellen Maschinenteilen erfolgreich eingesetzt. Vgl. HINKLE/TOOMEY (1995), S. 65 ff. In der Automobilindustrie wird das CBR-System COMPOSER u.a. zum Design von Montageabläufen eingesetzt. Vgl. PURVIS/PU (1998), S. 285 ff.
 - 16) Die Software Cassiopee unterstützt beispielsweise Flugzeugtechniker bei der Wartung von Flugzeugtriebwerken. Vgl. HEIDER (1996), S. 512 ff.
 - 17) Das Programm FM-Ultranet wird beispielsweise eingesetzt, um postoperative Symptome bei Transplantationspatienten zu diagnostizieren. Vgl. NILSSON/SOLLENBORN (2004), S. 179 f. Das System ICONS unterstützt Mediziner bei der Auswahl der Antibiotika-Therapie-Auswahl. Vgl. FREUDENTHALER (2008), S. 77 ff.
 - 18) Bei Compaq wird zum Beispiel SMART eingesetzt, ein CBR-Tool zur Abwicklung des technischen Supports. Vgl. CHEETHAM/WATSON (2006), S. 322 und Watson (1997), S. 93 ff.
 - 19) Beispielsweise verwendet der Reiseanbieter Berge & Meer Touristik in seinem Internet-Portal <http://www.reiseboerse.com> CBR-Retrieval zur Suche von Lastminute-Angeboten. Vgl. FREUDENTHALER (2008), S. 128 f. und BLANZIERI/EBRANATI (2000), S. 359.
 - 20) Vgl. SOMMER (2007), S. 83 ff.

Die Problemfälle im SCM sind zunächst hauptsächlich natürlichsprachig beschrieben, was für die Verwendung in einem CBR-System Modellierungsbedarf weckt, um die Komplexität bewältigen und „verstehen“ zu können.

Das wissenschaftliche Problem dieser Arbeit konstituiert sich darin, im Forschungsgebiet des Case-Based Reasonings die Lücke zwischen konventioneller quantitativer Problembeschreibung und der Handhabung qualitativer Probleme zu schließen und erstmals auf die Domäne Supply Chain Management zu beziehen.

Ziel der Arbeit ist es, im Rahmen des Verbundprojekts OrGoLo einen Prototyp für ontologiegestütztes Case-Based Reasoning in Supply Chains zu entwickeln, der als Idee oder Vorschlag in die Entwicklung eines Assistenztool-Prototyps für die Kollaborationsplattform eingehen kann.

1.3 Aufbau der Arbeit

Diese Arbeit verfolgt das Ziel, in der Anwendungsdomäne des Supply Chain Managements einen Prototyp für ontologiegestütztes Case-Based Reasoning mit qualitativem Retrieval zu entwickeln. Grundlegend werden zunächst die fundamentalen Themen Case-Based Reasoning und Ontologien vorgestellt und erläutert. Aufbauend auf dem Grundlagenwissen wird der Entwicklungsprozess in aufeinander aufbauenden Abschnitten von der Konzept-Phase bis zur Test-Phase beschrieben. Zum Schluss folgen ein Fazit zur Entwicklung des Prototyps sowie ein abschließender Ausblick auf zukünftige Entwicklungen.

Die Abbildung 1 veranschaulicht den Aufbau dieser Arbeit noch einmal.

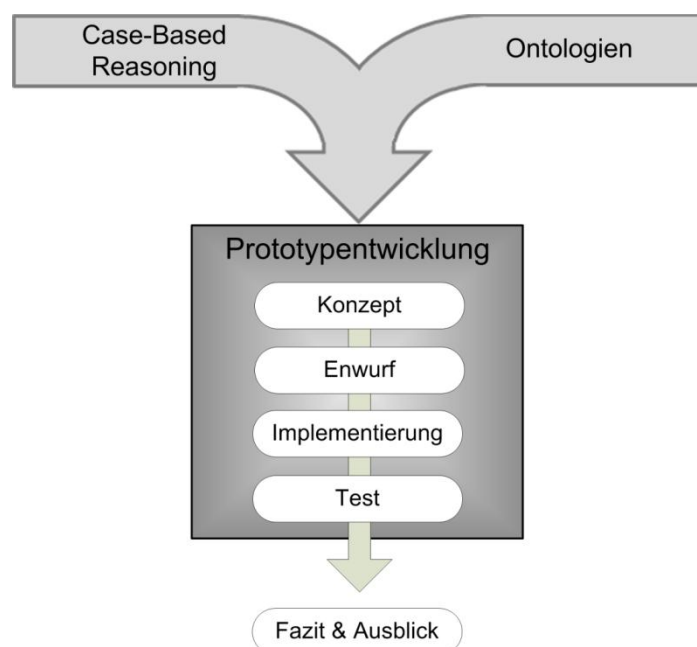


Abbildung 1: Aufbau der Arbeit

2 Case-based Reasoning

“Jede von mir gelöste Aufgabe wurde zum Muster, welches ich im Weiteren für die Lösung anderer Aufgaben nutzte.“

(René Descartes)

2.1 Die Intention des Case-based Reasonings

Die aus dem Forschungszweig der Künstlichen Intelligenz entsprungene Methode des Case-Based Reasonings (dt. fallbasiertes Schließen) unterscheidet sich wesentlich von anderen bedeutenden KI-Methoden und -Techniken. Anstatt Lösungswege ausschließlich aus generellem Domänenwissen (z. B. Regeln) herzuleiten setzt CBR zusätzlich das spezifische Wissen ein, das in der Vergangenheit praktisch erfahren wurde.²¹

Die philosophische Grundlage zu Case-Based Reasoning liefert LUDWIG WITTGENSTEIN in seinen „Philosophischen Untersuchungen“ von 1953.²² Wittgenstein kommt zu der Erkenntnis, dass sich Begriffe²³ nicht vollständig beschreiben und untereinander abgrenzen lassen, sich jedoch verschiedene Instanzen²⁴ eines Begriffs durch eine ganze Reihe von Ähnlichkeiten untereinander auszeichnen.²⁵ Durch diese Ähnlichkeiten können Begriffe unterschieden werden, und Instanzen ihnen zugeordnet werden.²⁶

In der Psychologie zeigen viele Untersuchungen²⁷, dass Menschen, wenn sie eine Aufgabe zu lösen haben, meistens an frühere ähnliche Aufgaben denken, und sich deren Lösung zur Bewältigung der neuen Aufgabe zur Hilfe nehmen. Dieser Gebrauch von Erfahrungen hat sich als geeignete Heuristik zur Lösung neuer Probleme herausgestellt. Der Prozess, Analogien zum Erfahrungswissen zu bilden, wird von vielen Kognitionswissenschaftlern als bedeutender Aspekt des menschlichen Lernens angesehen.

Inspiziert von der Idee, die menschliche Methodik zur Lösung von Problemen auf intelligente KI-Systeme zu übertragen, nutzen CBR-Anwendungen ein „Gedächtnis“ in Form einer Falldatenbank mit gespeicherten bereits gelösten Problemfällen. Diese „Erinnerungen“ sollen als Hilfe dienen, um neu auftretende Probleme zu lösen. Ein neues Problem wird bei Eingabe in das CBR-System mit den alten Fällen verglichen und anhand der ggf. modifizierten Lösungsbeschreibung des ähnlichsten Alt-Falls gelöst. Damit wird menschliches Verhalten imitiert – in vielen Situationen vergleichen wir Probleme mit ähnlichen Erfahrungen aus der Vergangenheit, um neue Problemsituationen auf ähnliche Weise bewältigen zu können.²⁸

21) Vgl. AAMODT/PLAZA (1994), S. 39.

22) Vgl. AAMODT/PLAZA (1994), S. 41.

23) Ein *Begriff* kann im CBR-Umfeld durch eine Problemsituation repräsentiert werden.

24) Als *Instanz* kann im CBR-Umfeld ein bestimmter Entscheidungsfall zu einer Problemsituation verstanden werden.

25) Vgl. WITTGENSTEIN/SCHULTE (2001), S. 787.

26) Dies entspräche einer konkreten Entscheidung in einer Problemsituation im CBR-Umfeld.

27) Vgl. ROSS (1989), S. 438 ff.

28) Vgl. KOLODNER (1992), S. 4.

2.2 Ein Fall im Case-based Reasoning

Ein Fall (engl. Case) beschreibt im Kontext des Case-Based Reasonings eine Problemsituation²⁹ und besteht aus:³⁰

1. einer Problembeschreibung, die die Situation zum Zeitpunkt des Problem-Auftretens beschreibt,
2. einer festgelegten oder abgeleiteten Lösung zum Problem, und
3. einem Ergebnis in Form einer Zustandsbeschreibung der Problemsituation nach Anwendung der Lösung.

Für gewöhnlich wird auf die Erfassung der Zustandsbeschreibung nach Lösung des Problems verzichtet, sodass ein Fall vereinfacht gesagt ein Paar aus einem Problem und seiner Lösung ist.³¹ Bei der Problemlösung ist zu beachten, dass nur solche mit einem zufriedenstellenden Ergebnis gespeichert werden, da der Fall sonst nicht als Beispiel zur Lösung neuer Fälle dienen kann.

Im Verlauf dieser Arbeit wird bei Fällen unterschieden zwischen den Begriffen:

1. „Neu-Fall“: für neuauftretende Fälle, die noch keine akzeptierte Lösungsbeschreibung enthalten, und
2. „Alt-Fall“: für bereits gelöste Fälle aus der Vergangenheit mit akzeptierter Lösungsbeschreibung.

2.3 Wissensrepräsentation in der Fallbasis

Das Herzstück eines Case-Based-Reasoning-Systems bildet die Falldatenbank. Hier werden die Alt-Fälle gespeichert und bilden als „Fall-Gedächtnis“ das Wissen eines CBR-Systems ab. Die Falldatenbank besteht aus der Ansammlung von Alt-Fällen sowie einer Menge von Zugangsprozeduren, die ein späteres Abrufen (engl. Retrieval) der Alt-Fälle ermöglichen sowie gelöste Neu-Fälle in die Struktur der Falldatenbank speichern sollen.³²

Die Struktur der Fallbasis ist, ebenso wie ihr Inhalt, von entscheidender Bedeutung für die Funktionalität eines CBR-Systems.³³ Die Struktur der Fallrepräsentation in der Fallbasis muss sowohl effektiv als auch effizient sein hinsichtlich der erfolgreichen Suche nach Alt-Fällen und der Integration neuer Fälle in die vorhandene Struktur. Diese Erfolgskriterien hängen ab von der Organisation und Indizierung der Falldatenbankstruktur.

29) Vgl. AAMODT/PLAZA (1994), S. 40.

30) Vgl. KOLODNER (1997), S. 147.

31) Vgl. PFUHL (2003), S. 6.

32) Vgl. KOLODNER (1997), S. 141.

33) Vgl. AAMODT/PLAZA (1994), S. 47.

Das Dynamic Memory Model³⁴ ist ein früher und vielzitatierter³⁵ Ansatz zur Organisation der Fallbasis im Case-Based Reasoning, sowie zur Modellierung der Gedächtnisstruktur beim Menschen. Das Dynamic Memory Model leitet sich aus der Prämisse ab, dass die Prozesse des Erinnerns, Verstehens, Erfahrens und Lernens im menschlichen Gehirn eng miteinander verknüpft sind und nicht nur separiert betrachtet werden können³⁶. Das menschliche Gehirn und damit auch das Gedächtnis unterliegen ständiger Veränderungen. Diese Veränderungen sind Resultate neuer Erfahrungen die bei der Konfrontation mit neuartigen Situationen auftreten, sowie der Fragen (und dazugehöriger Antworten), die im Bezug zu dieser Situation auftreten. Der Prozess des Verstehens funktioniert beim Menschen durch die Integration neuer spezieller Informationen in bereits vorhandenes generelles Wissen.

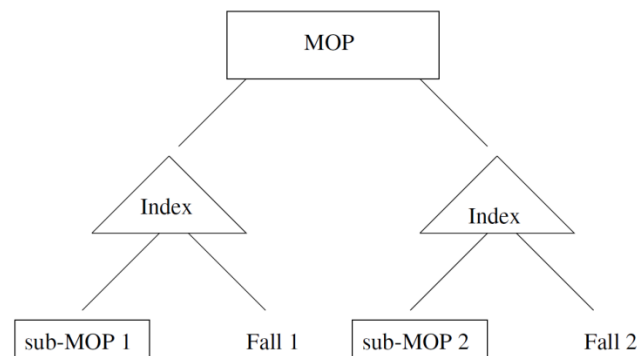


Abbildung 2: Memory Organization Packets

Im Dynamic Memory Model wird dieses generelle Wissen durch Memory Organization Packets (MOP) modelliert. Bei MOPs handelt es sich um Speicherstrukturen, die zum einen generelles Wissen speichern und zum anderen durch Indizes auf spezielle Erfahrungen (Alt-Fälle) verweisen und damit in einer komplexen Hierarchie die Verbindungen im Gedächtnis (Falldatenbank) organisieren. Wird nun ein neuer Fall in die Falldatenbank gespeichert, muss zunächst entschieden werden, an welcher Stelle er in der Baumstruktur (siehe Abbildung 2) des Dynamic Memory Model eingefügt werden soll. Zunächst werden anhand der im neuen Fall definierten Indizes die Pfade durch die Speicherstruktur durchlaufen und schließlich an dem spezifischsten MOP³⁷ eingefügt. Für das Einfügen gibt es zwei Möglichkeiten:³⁸

1. Wenn an der einzufügenden Stelle (MOP) bereits ein Alt-Fall indiziert ist, so wird dieser mit dem Neu-Fall verglichen und die Ähnlichkeiten werden zu einem neuen MOP zusammengefasst. Somit wird im neu geschaffenen MOP generelles Wissen generiert. Das neue MOP ist eine Spezialisierung des übergeordneten MOPs.
2. Wenn an der einzufügenden Stelle noch kein Alt-Fall besteht, wird ein neuer Index erstellt, der auf den neuen Fall verweist.

34) Vgl. SCHANK (1999), S. 1 f.

35) Vgl. AVRAMENKO/KRASLAWSKI (2008), S. 53; SMYTH et al. (2000), S. 301 ff.; CONWAY (1997), S. 238; AAMODT/PLAZA (1994), S. 41.

36) Vgl. KOLODNER (1997), S. 101.

37) Wird ein mit MOPs strukturiertes Dynamic Memory Model als baumartige Struktur betrachtet, so werden die MOPs umso spezifischer, je tiefer die Ebenen des Baums werden. Die Wurzel des Baums ist das generellste MOP.

38) Vgl. RIESBECK/SCHANK (1989), S. 36 f.

Das Dynamic Memory Model wurde erstmals von KOLODNER im „Computerized Yale Retrieval and Update System“ (CYRUS) in einem CBR-System eingesetzt.³⁹ In CYRUS werden Ereignisse im beruflichen Leben des ehemaligen US-amerikanischen Außenministers CYRUS R. VANCE gespeichert und zum Retrieval bereitgestellt.

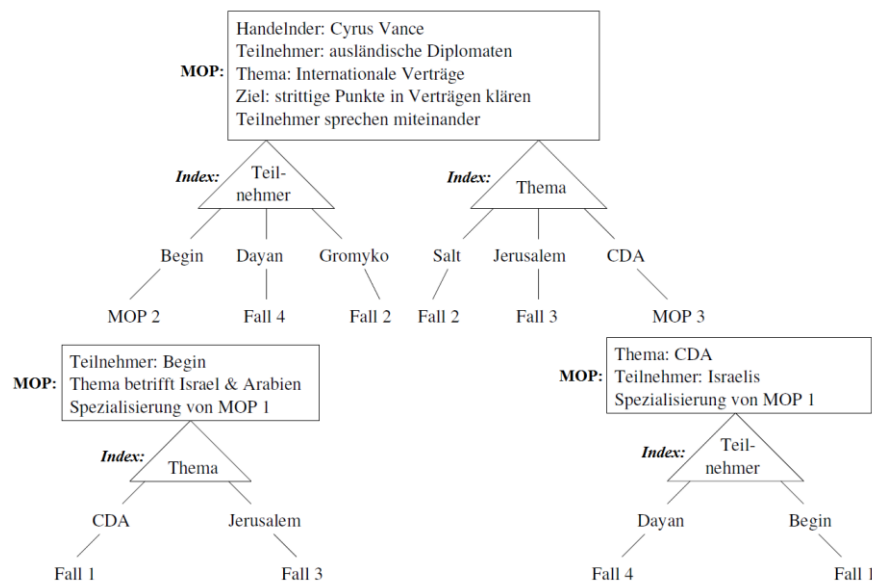


Abbildung 3: Ausschnitt aus dem Dynamic Memory Modell von CYRUS⁴⁰

Zur Veranschaulichung der Speicherstruktur in CYRUS zeigt Abbildung 3 anhand von vier Beispielfällen mit MOPs (MOP1-MOP3) und Indizes („Teilnehmer“ und „Thema“), wie die Speicherstruktur in einem Dynamic Memory Modell aussehen kann.

In Bezug auf Schlussfolgerungen (engl. Reasoning) bietet das Dynamic Memory Model durch die verschiedenen Pfade zu Alt-Fällen eine Vorauswahl. Zu jedem identifizierbaren Merkmal eines Falls kann eine Instanz des Falls in der Baumstruktur existieren (in Abbildung 3 existieren je zwei Instanzen zu den Fällen 1-4). Werden nun die spezifischsten MOPs (in Abbildung 3 MOP2 und MOP3) und die zugehörigen Indizes betrachtet, sind dort die ähnlichsten Fälle zu finden. Aus Sicht von Fall 1 sind somit die Fälle 3 und 4 Kandidaten für den ähnlichsten Fall. Die genauere Auswahl eines geeigneten Falls ist dann die Aufgabe einer geeigneten Retrieval-Technik.

2.4 Anwendungsformen des Case-based Reasonings

2.4.1 Strukturelles Case-based Reasoning

Aufgrund der vielfältigen Einsatzmöglichkeiten von Case-Based Reasoning in der Praxis (Help-Desk Support, Design & Konstruktion etc.) ergeben sich sehr unterschiedliche Anforderungen und Herangehensweisen an den Aufbau von CBR-Systemen im Hinblick auf die Art der Wissensgewinnung. In der Literatur lassen sich drei wesentliche Ansätze finden,⁴¹ die in den folgenden Abschnitten erläutert werden.

39) Vgl. KOLODNER (1997), S. 121 ff.

40) Vgl. STEIN et al. (2007), S. 6.

41) Vgl. BERGMANN/SCHAAF (2003), S. 609, PFUHL (2003), S. 18 f. und BERGMANN ET AL. (1999), S. 21.

Beim Strukturellen Case-Based Reasoning (SCBR) handelt es sich um die am häufigsten verwendete Anwendungsform von CBR.⁴² Hierbei werden die Merkmale von Fällen mithilfe von Attribut-Wert-Paaren modelliert. Ein Fall enthält demnach mehrere Attribute, denen jeweils ein Wert zugewiesen werden kann. Ein Attribut besteht aus einer Bezeichnung A und einem zulässigen Wertebereich (Domäne DOM), d.h. der Menge der Werte, die ein Attribut annehmen kann. Ein Fall kann somit im SCBR immer als Attribut-Wert-Vektor dargestellt werden.⁴³

$$Fall = \begin{pmatrix} a_1 \\ \dots \\ a_n \end{pmatrix}, \text{ für } a_i \in DOM(A_i), i = 1, \dots, n \quad (2.1)$$

Die Attribute gehören jeweils zu einem bestimmten Datentyp.⁴⁴ Dieser kann beispielsweise den verschiedenen Skalenniveaus zuordenbar sein (Nominal-Skala, Ordinal-Skala, Intervall-Skala und Ratio-Skala) und gilt dann als „einfaches Attribut“⁴⁵. Eine Erweiterung stellt die Verwendung von „relationalen Attributen“ nach dem Vorbild der Objektorientierten Programmierung dar.

Als Beispiel für Strukturelles Case-Based Reasoning modellierten BERGMANN/SCHAAF (2003) das Wissen über Computerhardware. Einfache Attribute stellen hier z.B. „Lese-/Schreibgeschwindigkeit von Festplatten“ und „Zugriffszeiten“ dar, denen Millisekunden-Werte zugeordnet werden. Das Attribut „Speichermedium“ ist relational und kann auf die Objekte „Hard Disk“, „CD-ROM“, „CD-RW“ oder „Optical Storage Device“ verweisen, die wiederum bestimmte Attribute (z.B. Herstellername) besitzen (siehe Abbildung 4).

C1	<table border="1"> <thead> <tr> <th>Hard Disk</th> </tr> </thead> <tbody> <tr> <td>manufacturer: Samsung</td> </tr> <tr> <td>capacity: 20.0</td> </tr> <tr> <td>access time: 9.5</td> </tr> <tr> <td>interface: IDE</td> </tr> </tbody> </table>	Hard Disk	manufacturer: Samsung	capacity: 20.0	access time: 9.5	interface: IDE	C2	<table border="1"> <thead> <tr> <th>CD-ROM</th> </tr> </thead> <tbody> <tr> <td>manufacturer: Sony</td> </tr> <tr> <td>interface: IDE</td> </tr> <tr> <td>read-speed: 48</td> </tr> </tbody> </table>	CD-ROM	manufacturer: Sony	interface: IDE	read-speed: 48
Hard Disk												
manufacturer: Samsung												
capacity: 20.0												
access time: 9.5												
interface: IDE												
CD-ROM												
manufacturer: Sony												
interface: IDE												
read-speed: 48												
C3	<table border="1"> <thead> <tr> <th>CD-RW</th> </tr> </thead> <tbody> <tr> <td>manufacturer: Teac</td> </tr> <tr> <td>interface: IDE</td> </tr> <tr> <td>read-speed: 40</td> </tr> <tr> <td>write-speed: 16</td> </tr> </tbody> </table>	CD-RW	manufacturer: Teac	interface: IDE	read-speed: 40	write-speed: 16	C4	<table border="1"> <thead> <tr> <th>Optical Storage Device</th> </tr> </thead> <tbody> <tr> <td>interface: IDE</td> </tr> <tr> <td>read-speed: 56</td> </tr> </tbody> </table>	Optical Storage Device	interface: IDE	read-speed: 56	
CD-RW												
manufacturer: Teac												
interface: IDE												
read-speed: 40												
write-speed: 16												
Optical Storage Device												
interface: IDE												
read-speed: 56												

Abbildung 4: Werte des relationalen Attributs "Speichermedium"⁴⁶

2.4.2 Textbasiertes Case-based Reasoning

Das Textbasierte Case-Based Reasoning (TCBR) unterscheidet sich wesentlich vom zuvor beschriebenen SCBR, da die Wissensrepräsentation hier kaum Struktur aufweist. Es handelt sich bei diesen Fällen um reine Texte, die den Sachverhalt eines Problems in einem Anwendungsfeld beschreiben. Typische Anwendungen für TCBR sind die häufig im Kundensupport anzutreffenden

42) Vgl. PFUHL (2003), S. 19.

43) Vgl. BERGMANN ET AL. (1999), S. 25.

44) Vgl. BERGMANN et al. (1999), S. 26 f.

45) Vgl. BERGMANN/SCHAAF (2003), S. 610 ff.

46) Vgl. BERGMANN/SCHAAF (2003), S. 613.

FAQ (Frequently Asked Questions). Die Fälle können vor der Einspeisung in die Fallbasis noch mit Zusatzinformationen versehen werden (Header), um zumindest einen geringen Grad an Strukturiertheit zu erlangen.

Vorteil des TCBR ist der geringe Aufwand bei der Akquirierung von Fällen, die im Idealfall ohne Anpassung in die Fallbasis integriert werden können. Der geringe Strukturierungsgrad zieht jedoch einige Probleme nach sich. Zum einen wird ohne erkennbare Ordnung in der Speicherstruktur der Fallbasis das Retrieval bei einer größeren Menge an Fällen unüberschaubar.⁴⁷ Zum anderen ist hier (im Gegensatz zum SCBR) ein wesentlich höherer Planungsaufwand vonnöten, um geeignete Retrieval-Techniken zu entwickeln, die den Anforderungen an ein CBR-System gerecht werden.⁴⁸

2.4.3 Dialogorientiertes Case-based Reasoning

Beim Dialogorientierten Case-Based Reasoning (engl. Conversational Case-Based Reasoning, CCBR) findet eine Konversation zwischen dem CCBR-System und einem Anwender statt. Der Anwender kann direkt ein Kunde mit einem Problem sein oder ein zwischengeschalteter Anwender, der mit dem System vertraut ist und als Vermittler dient⁴⁹ (z.B. Call Center Agent). Ziel beim CCBR ist es, mithilfe einer vordefinierten Menge an Fragen ein Ausgangsproblem zu erfassen und zu lösen. Das CCBR-System gibt dem Anwender die Fragen vor, die er beantworten muss um sein Problem zu beschreiben. Die Fallbeschreibung eines neuen Falls besteht dann aus eine Liste von Frage-Antwort-Paaren.

Die Fallbasis wird von Experten künstlich angelegt und enthält Fälle, die jeweils aus einer frei formulierten Problembeschreibung, einer Problemspezifikation sowie einer Handlungsempfehlung zur Lösung des Problems bestehen.⁵⁰ Die Problemspezifikation enthält eine Liste von Frage-Antwort-Paaren und wird zur Ähnlichkeitsbestimmung mit einem neu auftretendem Fall verwendet. Während das CCBR-System einen Dialog mit dem Anwender führt, wird durch die daraus akquirierten Frage-Antwort-Paare eine Rangliste mit den ähnlichsten Fällen aus der Fallbasis erstellt.

Anwendungsgebiete für CCBR liegen im Call-Center-Betrieb und Help Desk Support. Ein Beispiel aus dem Logistikbereich bietet die in Österreich entwickelte „Dynamic Logic Engine“⁵¹, bei der durch CCBR eine interaktive Selektion von vordefinierten Handlungsempfehlungen für Ausnahmesituationen stattfindet.

2.5 Case-based-Reasoning-Zyklus

2.5.1 Einleitung

AAMODT & PLAZA beschrieben 1994 die Struktur von Case-Based-Reasoning-Systemen als einen kontinuierlich ablaufenden Zyklus von vier Prozessen: Retrieve, Reuse, Revise und Retain. Dieses

47) Vgl. BERGMANN ET AL. (1999) S. 21.

48) Vgl. KUCKARTZ (2010), S. 108 f.

49) Vgl. AHA/BRESLOW (1997), S. 267 f.

50) Vgl. PFUHL (2003), S. 21.

51) Vgl. WEBER/WILD (2005), S. 137.

auch R⁴-CBR-Zyklus genannte klassische Prozessmodell (siehe Abbildung 5) beschreibt die wesentlichen Phasen des Case-Based Reasonings und ist in Wissenschaft und Praxis anerkannt.⁵²

In der Literatur finden sich noch weitere speziellere Ansätze, den Case-Based Reasoning-Zyklus zu modellieren. Die vier Phasen Retrieve, Reuse, Revise und Retain lassen sich mit „Repartition“ noch um ein fünftes „R“ erweitern. Repartition (dt.: Neuzuweisung) kann besonders bei riesigen Datenmengen nötig werden um die Komplexität der Fallbasis zu vermindern und folglich ein effizientes Retrieval zu gewährleisten.⁵³ Um dies zu erreichen, wird die Falldatenbank laufend unter Berücksichtigung der Case-Analogien neu strukturiert.

Zwei weitere Ansätze gehen noch einen Schritt weiter, und unterteilen den CBR-Zyklus in zwei ineinandergreifende Teile Anwendungszyklus und Wartungszyklus.⁵⁴ Für die Zwecke dieser Arbeit ist das klassische Modell von AAMODT & PLAZA ausreichend, sodass dieses im Weiteren als grundlegendes Prozessmodell verwendet wird.

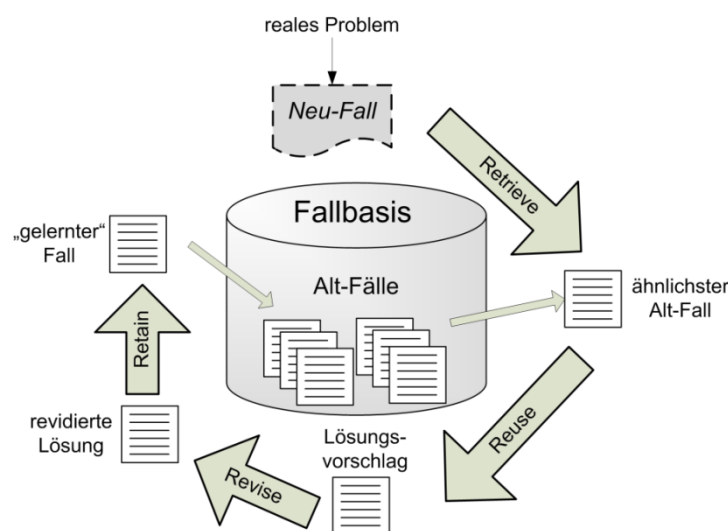


Abbildung 5: CBR-Zyklus⁵⁵

Im Folgenden werden die vier Hauptphasen des CBR-Prozessmodells näher vorgestellt, wobei der Hauptfokus auf das Case Retrieval gelegt wird, da hierauf der Schwerpunkt dieser Arbeit gelegt wird.

2.5.2 Retrieve

Zu Beginn gelangt ein zu untersuchendes Problem als neuer Fall in den Prozess-Zyklus. Unter der Annahme, dass ähnliche Probleme auch auf ähnliche Art und Weise gelöst werden können, wird

52) Vgl. AVRAMENKO/KRASLAWSKI (2008), S. 60; GÖRZ/ROLLINGER/SCHNEEBERGER (2003), S. 410 f.; PFUHL (2003), S. 13.

53) Vgl. FINNIE/SUN (2002), S. 60 f.

54) In GÖKER/ROTH-BERGHOFER (1999), S. 143 f., wird ein Wartungszyklus beschrieben, der aus den Teilprozessen *Recycle*, *Refine* und *Retain* besteht und im Prozess *Recycle* eine Schnittstelle mit dem klassischen CBR-Zyklus hat. Alternativ dazu wird in ROTH-BERGHOFER (2003) ein Wartungszyklus vorgestellt, der die Teilprozesse *Restore*, *Revise* und *Retain* enthält und zusätzlich zum 1999er Ansatz eine Unterteilung der Fallbasis in zwei Wissenscontainer vorsieht.

55) Eigene Darstellung in Anlehnung an AAMODT/PLAZA (1994), S. 45.

mithilfe der formalen Problembeschreibung des Neu-Falls in der Fallbasis nach einem möglichst ähnlichen Alt-Fall gesucht. Dieser Prozess des Durchsuchens und Abfragens der Fallbasis wird Retrieval genannt.

Die Besonderheit beim Retrieval im CBR ist die Notwendigkeit einer „intelligenten“ Heuristik für die Suche nach für die Wiederverwendung geeigneten Alt-Fällen. Die Form des Retrievals ist bei einer CBR-Fallbasis im Gegensatz zum präzisen Retrieval in einer relationalen Datenbank mittels SQL-Suchanfragen eher vage, es wird nicht nach einem Fall gesucht, der genau bestimmte Parameter erfüllt, sondern nach einem möglichst ähnlichen Fall, wobei das Maß der „Ähnlichkeit“ von der gewählten Heuristik abhängt.

Wenn ein Fall ausschließlich quantitative – also in Zahlen ausdrückbare – Eigenschaften besitzt, bieten sich einfache Techniken, wie beispielsweise das Nearest-Neighbour-Verfahren⁵⁶ zur Ermittlung der Ähnlichkeiten beim Retrieval an. Hierbei wird ein Neu-Fall anhand seiner Attribute paarweise mit Alt-Fällen aus der Fallbasis verglichen.

$$sim(x, y) = \sum_{i=0}^I sim_a(a_i^x, a_i^y) * w_i \quad (2.2)$$

Zunächst wird für jedes Attribut a_i mit dem attributspezifischen Index i ($i = 1, \dots, I$) zweier zu vergleichender Fälle x und y der Ähnlichkeitsgrad sim_a (z.B. mit Werten zwischen 0 und 1) ermittelt und anschließend mit einem Koeffizienten w_i gewichtet. Die addierten Vergleichsergebnisse der einzelnen Attribute bilden schließlich das Maß $sim(x, y)$ der Ähnlichkeit der beiden untersuchten Fälle (globaler Ähnlichkeitsmaßstab).

Mit steigender Größe der Fallbasis kann das Nearest-Neighbour-Verfahren zu aufwendig werden, da sich mit wachsender Anzahl der zu vergleichenden Alt-Fälle auch der Berechnungsaufwand erhöht.

Die Gruppe der Induktions-Verfahren bietet ein schnelleres Retrieval als das zuvor vorgestellte Verfahren. Das Retrieval durch Induktion erfolgt mithilfe von Entscheidungsbäumen oder Regeln. Durch einen Induktions-Algorithmus⁵⁷ wird die Falldatenbank analysiert, um einen Entscheidungsbaum zu erstellen, der die Alt-Fälle so indiziert, dass der Baum eine binäre Struktur erhält, die in Bezug auf ein Zielattribut möglichst gleichmäßig verteilt ist.⁵⁸ Im Beispiel der Einschätzung von Privatkrediten in einem Kreditinstitut in Abbildung 6 wird das Zielattribut durch den Kreditstatus verkörpert. Durch die Aufteilung des Entscheidungsbaums anhand der Attribute „Rückzahlung“, „Arbeitsstatus“ und „Einkommen“ wird die Fallbasis in zwei Hälften geteilt.

56) Vgl. PFUHL (2003), S. 64 f. und WATSON (1997), S. 23 ff.

57) Der bekannteste Induktions-Algorithmus ist ID3, vgl. WATSON (1997), S. 28 ff.

58) Das Zielattribut verkörpert hier die gesuchte Lösung eines Falls im CBR, vgl. WATSON (1997), S. 29.

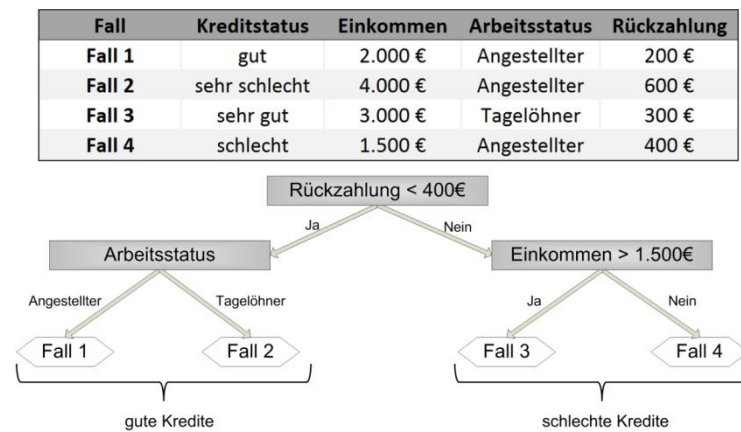


Abbildung 6: Entscheidungsbaum beim Induktiven Retrieval⁵⁹

Wird nun ein Neu-Fall in diesem Beispiel untersucht, so werden nur maximal zwei Attribute überprüft (z.B. „Rückzahlung“ und „Arbeitsstatus“), um den ähnlichsten Alt-Fall zu ermitteln. Im Vergleich zum Nearest-Neighbour-Verfahren ist das induktive Retrieval (auch bei zunehmender Größe der Fallbasis) wesentlich schneller, birgt jedoch auch den Nachteil, dass nach jedem Hinzufügen eines neuen Falls der Entscheidungsbaum neu berechnet werden muss.⁶⁰

2.5.3 Reuse

Nachdem mindestens ein (dem Neu-Fall möglichst ähnlicher) Alt-Fall aus der Fallbasis abgerufen wurde, geht es im Reuse-Teilprozess darum, die Lösungsbeschreibung des Alt-Falls zur Generierung einer Lösung für den Neu-Fall wiederzuverwenden. Die einfachste Möglichkeit ist die Null-Adaption⁶¹ der alten Lösung an das neue Problem, d.h. die alte Lösung ohne Änderung zu kopieren und auf den Neu-Fall anzuwenden. Die Null-Adaption ist nur anwendbar, wenn die Lösungen der Fälle allgemein sehr simpel strukturiert sind. Ist das nicht der Fall, muss eine Adaption der alten Lösung geschehen, um sie auf den Neu-Fall anwenden zu können.

Bei der Adaption im Reuse-Teilprozess von Case-Based Reasoning wird unterschieden zwischen struktureller und abgeleiteter Adaption.⁶² Strukturelle Adaption wendet bestimmte Adaptions-Regeln an, um die Lösungen der Alt-Fälle direkt zu ändern, beispielsweise werden einzelne Parameter von Alt- und Neu-Fall verglichen und davon abhängige Parameter in der Lösung verändert. Abgeleitete Adaption verwendet den Lösungsweg (z.B. Regeln oder Formeln), der zur Generierung der Lösung des Alt-Falls angewandt wurde, und benutzt die gleichen Regeln und Formeln, um den Neu-Fall zu lösen. Hierfür ist es in einem CBR-System notwendig, für jeden Fall zusätzlich zur Lösungsbeschreibung Informationen über den Lösungsweg abzuspeichern.

Die automatisierte Adaption im Reuse-Teilprozess von Case-Based Reasoning ist für komplexe Einsatzgebiete noch nicht ausgereift genug, sodass in den meisten CBR-Systemen der menschliche Anwender die Adaption übernimmt.⁶³

59) Eigene Darstellung in Anlehnung an WATSON (2001), S. 29 ff.

60) Vgl. PFUHL (2003), S. 66.

61) Vgl. WATSON (2001), S. 34 und AAMODT/PLAZA (1994), S. 51.

62) Vgl. WATSON (1997), S. 34 f.

63) Vgl. LEAKE (1996), S. 9.

2.5.4 Revise

Der im *Reuse* erstellte Lösungsvorschlag wird in Bezug auf seine Anwendungseignung evaluiert, indem er entweder in der realen Anwendungsumgebung des Problemfalls angewandt wird, von einem menschlichen Experten evaluiert wird oder durch Simulation geprüft wird.⁶⁴ Wenn der neue Lösungsvorschlag einen dieser Tests besteht, geht der Neu-Fall in den Teilprozess Retain über. Sollte die Evaluation nicht zum gewünschten Erfolg führen, wird der Lösungsvorschlag dementsprechend revidiert und angepasst, bis eine akzeptable Lösung erreicht wird.

Die Revidierung eines gescheiterten Lösungsvorschlags besteht aus zwei Handlungsschritten⁶⁵. Zunächst gilt es, die Fehler in der vorliegenden Lösung zu identifizieren und Erklärungen für deren Auftreten zu suchen und zu generieren. Anschließend gilt es, anhand der Fehlererklärungen die Lösung zu reparieren, sodass die Fehler nicht mehr auftreten. Dies kann beispielsweise durch Hinzufügen von Arbeitsschritten, die das Auftreten der Fehler verhindern, geschehen. Die revidierte Lösung wird nun so lange evaluiert und ggf. erneut repariert, bis eine erfolgreiche Lösung des Neu-Falls erzielt wird.

2.5.5 Retain

In diesem Teilprozess findet der Lerneffekt des Case-Based Reasonings statt. Die zuvor erfolgreich evaluierte und getestete Lösung des Neu-Falls wird in die Falldatenbank gespeichert, um zukünftig als Vorlage für die Lösung ähnlicher Probleme hilfreich sein zu können. Zusätzlich können Informationen gespeichert werden, die Aussagen über die ggf. durchgeführten Modifizierungen sowie aufgetauchte Fehler ermöglichen. Im CBR-Teilprozess Retain werden Entscheidungen darüber getroffen, in welcher Form der Neu-Fall gespeichert werden soll, welche Indizes zum späteren Retrieval erstellt werden sollen (siehe Kapitel 2.3) und wie die Fallbasis ggf. passend zum neu erlangten Wissen angepasst werden muss. Nach der Speicherung des Neu-Falls in die Fallbasis hat das CBR-System diesen Fall „gelernt“. Indem die ursprüngliche Problembeschreibung des Neu-Falls erneut als Fall ins Retrieval eingegeben wird, kann die Funktionsweise des CBR-Systems getestet werden: Der soeben gespeicherte Fall muss im positiven Testfall als ähnlichster Alt-Fall vorgeschlagen werden.⁶⁶

3 Ontologien

3.1 Einführung in das Themenfeld „Ontologien“

3.1.1 Einleitung

Die im vorigen Abschnitt vorgestellte Methode des Case-Based Reasonings wird im Rahmen dieser Arbeit in der wissensintensiven Anwendungsdomäne des Supply Chain Managements eingesetzt. Ausschnitte aus der realen Welt der Supply Chains müssen in eine Darstellungsform überführt wer-

64) Vgl. PFUHL (2003), S. 14.

65) Vgl. AAMODT/PLAZA (1994), S. 52.

66) Vgl. AAMODT/PLAZA (1994), S. 52 f.

den, die vom Computer (CBR-System) „verstanden“ und für Schlussfolgerungen (engl. Reasoning) genutzt werden kann. Das für die Aufgabenbereiche im Supply Chain Management relevante Wissen muss von allen beteiligten Akteuren (z.B. CBR-System und verschiedenste menschliche Anwender) übereinstimmend verstanden werden, da differente Verständnisse derselben Gegebenheiten den Austausch von Informationen und die Zusammenarbeit der Akteure erschwert.⁶⁷ Um ein einheitliches Verständnis von Konzepten zu erzielen, bedarf es eines Instruments zur Spezifizierung und Klassifizierung des vorhandenen Wissens.

Die Technik der Ontologien erlaubt eine systematische Darstellung und Analyse des vorhandenen Wissens auf semantischer Ebene⁶⁸ und wird daher als Technik zur Wissensrepräsentation im hier behandelten Case-Based Reasoning Ansatz verwendet.

In diesem Abschnitt werden grundlegende Prinzipien von Ontologien vorgestellt, die notwendig sind, um die Rolle von Ontologien in einem Case-Based-Reasoning-System zu verstehen. Zunächst werden Ursprünge und unterschiedliche Verständnisweisen von Ontologien dargelegt. Im Anschluss daran wird die „Konzeptualisierung“ von Ontologien vorgestellt. Abschließend wird erläutert, wie Ontologien formuliert werden können.

3.1.2 Philosophischer Ursprung

Die Thematik der Ontologien⁶⁹ ist keine Neuschöpfung der Informatik – die Ursprünge finden sich bereits im antiken Griechenland und sind philosophischer Natur. ARISTOTELES befasste sich in seinem Werk „Metaphysik“ mit der Frage nach dem Wesen des „Seiendem“⁷⁰. Entgegen der Vorstellung seines Lehrers PLATON, der die Ideenlehre⁷¹ begründete, glaubte ARISTOTELES daran, dass das „Wesen“ der Dinge in ihnen selbst liegt, und die Dinge untersucht werden müssten, um zu ihrem innersten „Wesen“ vordringen zu können. Im Gegensatz zur experimentellen Physik, die die Realität stets unter einer bestimmten realen Perspektive zu ergründen und modellieren versucht, liegt der Fokus der Ontologie in der Natur der Dinge an sich, unabhängig davon ob Dinge real existieren oder nicht⁷². Hierfür befasst sich die Ontologie mit der Bildung von Kategorien, die als möglichst allgemein definierte Klassen realer Phänomene die realen Dinge systematisch zu strukturieren versuchen⁷³.

67) Vgl. STUCKENSCHMIDT (2009), S. 6 ff.

68) Vgl. ZELEWSKI (2005), S. 127.

69) Im Folgenden werden die voneinander abzugrenzenden Begriffe „Ontologie“ und „Ontologien“ verwendet, wobei der Singular die Ontologie als Philosophie und der Plural das moderne zweckbezogene Ontologieverständnis aus Sicht der (Wirtschafts-)Informatik und Betriebswirtschaftslehre versteht. Vgl. ZELEWSKI (2005), S. 141 f. Das Wort „Ontologie“ ist zusammengesetzt aus dem Griechischen „ontos“ (dt. Sein) und „logos“ (dt. Wort) und wurde erst seit dem frühen 17. Jahrhundert von RUDOLF GÖCKEL eingeführt (vgl. JANSEN (2008a), S. 229). ARISTOTELES verwendete das Wort „kategoria“ (dt. Kategorie) um Dinge zu klassifizieren, vgl. JANSEN (2008b), S. 90 ff.

70) Vgl. GUARINO/OBERLE/STAAB (2009), S. 1 und ZELEWSKI (2005), S. 118.

71) Platon vertrat die Ansicht, dass „Ideen“ (oder auch „Formen“) als real existierende und unveränderliche Prinzipien das Wesen der Dinge vorgeben und beschreiben. Dabei wird die Idee von dem real existierenden Dingen getrennt betrachtet – die Idee ist im Ding allgegenwärtig und das Ding ist Teil der Idee. Vgl. ERLER (2006) S. 146 ff.

72) Vgl. GUARINO/OBERLE/STAAB (2009), S. 2.

73) Vgl. ZELEWSKI (2005), S. 118 f.

3.1.3 Modernes Ontologieverständnis

Während sich die Philosophie mit einer Ontologie als „Seinslehre“ beschäftigt hat, entstand im 20. Jahrhundert in den wissenschaftlichen Disziplinen der Betriebswirtschaftslehre und der (Wirtschafts-)Informatik, speziell im Bereich der Künstlichen Intelligenz, ein neues Ontologieverständnis.⁷⁴

Im Gegensatz zur allgemeinen philosophischen „Seinslehre“ befasst sich die moderne Wissenschaft mit Ontologien im Plural. Der Fokus der neueren wissenschaftlichen Arbeiten liegt nicht mehr darauf, abstrakte Aussagen über „das Sein“ an sich zu machen, sondern von Menschen geschaffene, sprachlich verfasste Artefakte (Ontologien) zu erstellen, die betriebswirtschaftlichen Zwecken dienen und zur Erkenntnisgewinnung und Gestaltung des betrieblichen Wissensmanagements instrumentalisiert werden können.⁷⁵

Diese Veränderung des allgemeinen Ontologieverständnisses ist auf Forschungsarbeiten in der Künstlichen Intelligenz zurückzuführen. Etwa in den 1980er Jahren kam dort ein besonderes Interesse für die Frage auf, wie die Kommunikation zwischen (sowohl menschlichen als auch künstlichen) Akteuren aufeinander abgestimmt werden kann, wenn der Informationsaustausch einen gemeinsamen Wissenshintergrund erfordert.⁷⁶ Insbesondere dann, wenn die betrieblichen Prozesse wissensintensiv und stark abhängig von der funktionierenden Zusammenarbeit verschiedener Akteure (mit ggf. unterschiedlichen Interpretationen desselben Wissenshintergrunds) sind – wie es im Supply Chain Management der Fall ist – wird ein Instrument benötigt, das in der Lage ist, das Wissen zu strukturieren und Unterschiede in dessen Interpretation zu minimieren.

Für den Begriff der „Ontologien“ im neueren Verständnis liegen verschiedene Interpretationen und Definitionen vor. Aus der Sicht der KI-Forschung wird mit einer Ontologie vorrangig die Spezifizierung eines Vokabulars für sprachliche Artefakte verbunden.⁷⁷ Dieses Vokabular definiert eine Sammlung von zulässigen Begriffen zur einheitlichen Beschreibung einer von mehreren Akteuren verwendeten Wissensdomäne.⁷⁸ Zusätzlich dazu enthält das Vokabular Annahmen über die Bedeutung der Begriffe sowie Regeln die die Relation der Begriffe untereinander beschreiben.

Die am weitesten verbreitete Definition⁷⁹ von Ontologien stammt von GRUBER aus dem Jahr 1993. Genauer betrachtet bietet GRUBER zwei offene Definitionsansätze. Zum einen beschreibt er Ontologien ähnlich dem Verständnis aus Sicht der Künstlichen Intelligenz als:

„(...) specification of a representational vocabulary for a shared domain of discourse – definitions of classes, relations, functions, and other objects.“⁸⁰

74) Vgl. DITTMANN/SCHÜTTE/ZELEWSKI (2003), S. 3 f.

75) Vgl. ZELEWSKI (2005), S. 121.

76) Vgl. DITTMANN/SCHÜTTE/ZELEWSKI (2003), S. 2.

77) Vgl. Zelewski (2005), S. 143.

78) Vgl. GUARINO (1998), S. 4 und GRUBER (1993), S. 199.

79) Beispielsweise wird in folgenden Publikationen auf GRUBERS Definition von Ontologien (teilweise auch kritisch) eingegangen: GUARINO/OBERLE/STAAB (2009), S. 2; DITTMANN/SCHÜTTE/ZELEWSKI (2003), S. 4; FENSEL (2004), S. 2; PINTO/MARTINS (2004), S. 442; GUARINO (1998), S. 3 ff.; MAASS (2009), S. 106; STUCKENSCHMIDT (2009), S. 22; GAŠEVIĆ/DJURIĆ/DEVEDŽIĆ (2006), S. 46.

80) GRUBER (1993), S. 199.

Des Weiteren formuliert GRUBER die vielzitierte und allgemein gehaltene Ontologiedefinition „*an ontology is an explicit specification of a conceptualization*“,⁸¹ wobei er nicht festlegt, was er genau unter einer expliziten Spezifikation versteht. Er lässt beispielsweise offen, ob die Spezifikation ontologischer Begriffe an bestimmte formale Sprachen gebunden ist oder nicht. Auch aufgrund dieser Ungenauigkeit leitet ZELEWSKI aus den offenen Definitionsansätzen von GRUBER folgende Arbeitsdefinition ab:

„Eine Ontologie ist die explizite und formalsprachliche Spezifikation einer Konzeptualisierung von Phänomenen in einem Realitätsausschnitt, die von mehreren Akteuren zur Erfüllung ihrer Zwecke gemeinsam verwendet wird.“⁸²

Die Formulierung von ZELEWSKI schafft im Gegensatz zu der von GRUBER eine unmissverständliche Definition des Ontologiebegriffs und bildet daher die Grundlage für das Ontologieverständnis dieser Arbeit, wobei der Realweltausschnitt durch das Supply Chain Management repräsentiert wird.

3.1.4 Konzeptualisierung

Im Grunde liegt jeder Wissensbasis und darauf basierenden Systemen und Agenten eine Konzeptualisierung zugrunde. Unter einer Konzeptualisierung wird eine abstrakte und simplifizierte Sichtweise auf Phänomene eines Realweltausschnitts verstanden, die zu einem vorgegebenen Erkenntniszweck repräsentiert werden sollen⁸³. Dabei bestimmt der Erkenntniszweck, welche Aspekte der Realweltphänomene für die Konzeptualisierung relevant sind. Der Erkenntniszweck kann beispielsweise sein, den Entscheidungsprozess zum Versand von Gütern in einem Logistikunternehmen abzubilden und zu „verstehen“, wobei von vermeintlich irrelevanten Realweltphänomenen wie z.B. den Gehältern der Angestellten, abstrahiert werden kann, die in die Konzeptualisierung einfließen.

Die Konzeptualisierung lässt sich mathematisch in Form einer „extensionalen“⁸⁴ relationalen Struktur“ repräsentieren.⁸⁵ Die Konzeptualisierung C ist demnach ein 2-Tupel (D, R) aus einem Gegenstandsbereich D (engl. für „discourse“) und einer Menge von Relationen R auf diesen Gegenstandsbereich. Jedes Element aus R ist eine mathematische Relation, die sich auf eine Menge von Elementen aus D bezieht und somit eine Erweiterung des Gegenstandsbereichs darstellt (daher der Begriff „erweiterte relationale Struktur“).⁸⁶ Wird als Beispiel⁸⁷ ein Software-Unternehmen betrachtet, das 50.000 Mitarbeiter beschäftigt, denen eindeutige Personalnummern zugeteilt sind, so könnte eine Konzeptualisierung wie folgt aussehen:

- $C = (D, R)$

81) GRUBER (1993), S. 199 und GRUBER (1995), S. 907.

82) ZELEWSKI (2005), S. 145, die Definition lässt sich noch weiter verfeinern, vgl. ZELEWSKI (2005), S. 153.

83) Vgl. ZELEWSKI (1999), S. 4 und GRUBER (1993), S. 199.

84) Eine Extension eines Konzepts (hier *Relation*) ist die Menge aller Entitäten, die das Konzept erfüllen.

85) GUARINO/OBERLE/STAAB (2009), S. 3 ff.

86) Vgl. GUARINO/OBERLE/STAAB (2009), S. 3.

87) Das Beispiel ist angelehnt an das Beispiel GUARINO/OBERLE/STAAB (2009), S. 4 ff.

- $D = \{PN00001, \dots, PN50000\}$
- $R = \{Person, Entwickler, Manager, kooperierMit, berichtetAn\}$

Die unären Relationen *Person*, *Entwickler* und *Manager* klassifizieren die Elemente des Gegenstandsbereichs D , während die binären Relationen *kooperiertMit* und *berichtetAn* jeweils Bezüge zwischen zwei Elementen herstellen. Die Aussage $berichtetAn \supseteq \{(PN02374, PN23239)\}$ sagt aus, dass der Angestellte mit der Personalnummer $PN02374$ an denjenigen mit der Personalnummer $PN23239$ Bericht erstattet.

Mit der erweiterten relationalen Struktur lässt sich ein Realweltausschnitt jedoch nur statisch zu einem bestimmten Zeitpunkt beschreiben. Da eine statische Beschreibung der Realwelt nicht den Anforderungen an die Modellierung einer dynamischen Realwelt genügt, beschreibt GUARINO eine Konzeptualisierung als „intensionale relationale Struktur“⁸⁸. Eine intensionale Relation R' ist eine mathematische Funktion, die in Bezug auf eine Menge W an möglichen Realweltzuständen W auf eine extensionale Relation R abbildet, d.h. in Abhängigkeit von der Variable „Realweltzustand“ kann sich die Ergebnismenge der Relationen unterscheiden. Nach GUARINO wird eine Konzeptualisierung C durch ein 3-Tupel beschrieben. Im zuvor beschriebenen Anwendungsbeispiel könnte eine „intensionale relationale Struktur“ wie folgt aussehen:

- $C = (D, W, R')$
- $D = \{PN00001, \dots, PN50000\}$
- $W = \{w_1, w_2, \dots\}$
- $R = \{Person, Entwickler, Manager, kooperiertMit, berichtetAn\}$

Die Ergebnismengen der intensionalen Relationen $kooperiertMit(w_1)$ und $kooperiertMit(w_2)$ können beispielsweise unterschiedliche extensionale Relationen enthalten, was durch eine Veränderung der Gegebenheiten der Realwelt innerhalb eines bestimmten Zeitraums verursacht werden kann. Wenn zwei Betrachter der Struktur im selben Realweltzustand unterschiedliche Ergebnismengen einer intensionalen Relation erhalten – was z. B. an einer divergenten Interpretation der Bedeutung von „Kooperation“ liegen kann – so haben sie unterschiedliche Konzeptualisierungen von den Phänomenen des betrachteten Realweltausschnitts⁸⁹.

3.1.5 Spezifikation

Menschen haben in der Regel implizite Vorstellungen von Konzeptualisierungen⁹⁰. Die Aufgabe von Ontologien ist es, wie schon in den Definitionen von GRUBER und ZELEWSKI erwähnt, Konzeptualisierungen explizit zu spezifizieren, d.h. unausgesprochen verstandene implizite Vorstellungen auf eindeutige und klare Weise auszudrücken und festzulegen. Diese Spezifikation der Konzeptualisierung ist notwendig, damit verschiedene miteinander kommunizierende Akteure eine gemeinsam verwendete Sprache nicht unterschiedlich interpretieren.

88) Vgl. GUARINO/OBERLE/STAAB (2009), S. 5 ff.

89) Vgl. GUARINO/OBERLE/STAAB (2009), S. 6.

90) Vgl. GUARINO/OBERLE/STAAB (2009), S. 8.

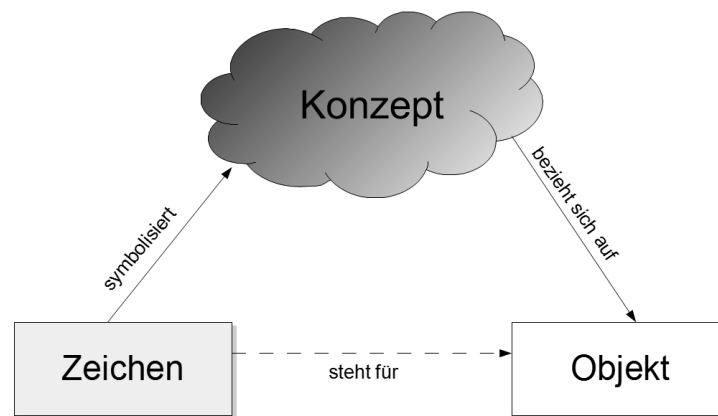


Abbildung 7: Semiotisches Dreieck⁹¹

Im semiotischen⁹² Dreieck (siehe Abbildung 7) wird der Zusammenhang von verwendeter Sprache (Zeichen oder auch Wörter), Konzepten und Objekten⁹³ in der Kommunikation veranschaulicht. Ausgangspunkt der Betrachtung sind Zeichen, die sich in einer bestimmten Darstellungsform – wie beispielsweise dem Wort *Person* – auf Objekte beziehen. Im Beispiel aus Abschnitt 3.1.4 steht das Wort *Person* für die Angestellten des Software-Unternehmens. Das Wort symbolisiert dabei ein Konzept, das mit der Verwendung des Wortes „gemeint“ ist. In der Kommunikation zwischen zwei Akteuren, in der der Sender sich mit Benutzung eines bestimmten Zeichens ausdrückt, versteht der Empfänger das Zeichen, so wie es vom Sender gemeint wurde, wenn er damit das gleiche Konzept assoziiert. Assoziiert der Empfänger allerdings ein anderes Konzept mit dem artikulierten Zeichen, kommt es zu Missverständnissen. Beispielsweise können zwei Gesprächspartner unterschiedliche Konzepte zum Wort *Person* haben: *Person* kann ein Synonym für *Mensch* sein (*Person = Mensch*), aber auch, wie in der Betriebswirtschaftslehre oder Rechtswissenschaft üblich, juristische Personen einbeziehen (*Person = {Mensch, GmbH, AG, ...}*).

Die explizite Spezifikation einer Konzeptualisierung soll bewirken, dass Konzepte eindeutig und unmissverständlich interpretiert werden. Spezifiziert werden kann auf zwei Arten⁹⁴: extensional und intensional. Die extensionale Spezifizierung erfordert eine Auflistung sämtlicher möglicher Relationen einer Konzeptualisierung zu allen möglichen Realweltzuständen. In den meisten Fällen ist dies unmöglich oder zumindest unpraktisch, so dass die intensionale Spezifizierung die effektivere Vorgehensweise darstellt.

Um eine Konzeptualisierung intensional zu spezifizieren, ist es notwendig, eine Sprache zu definieren, in der Relationen mithilfe von Axiomen (mathematische Grundsätze) beschrieben und eingeschränkt werden. Wird als Sprache zur Beschreibung einer Ontologie eine einfache Prädikatenlogik verwendet, so kann im Beispiel aus Abschnitt 3.1.4 durch eine Menge von fünf logischen Ausdrü-

91) Eigene Darstellung in Anlehnung an STUCKENSCHMIDT (2009), S. 7.

92) Die Semiotik ist die Lehre von Zeichen und deren Bedeutungen und untersucht seit Anfang des 20. Jahrhunderts den komplexen Zusammenhang der Abbildung von Objekten (Phänomenen der Realwelt) auf eine sprachliche Darstellung. Erste Überlegungen zu dieser Thematik finden sich allerdings bereits auch bei ARISTOTELES. Vgl. STUCKENSCHMIDT (2009), S. 7 f.

93) Unter dem Begriff „Objekt“ wird hier ein Phänomen eines Realweltausschnitts verstanden.

94) Vgl. GUARINO/OBERLE/STAAB (2009), S. 8.

cken O_1 bis O_5 mit steigender Präzision eine Ontologie spezifiziert werden. Dabei werden die Relationen miteinander in logischen Bezug gebracht, um eine semantische Ebene zu schaffen:⁹⁵

- Taxonomie:
 - $O_1 = \{ \text{Entwickler}(x) \rightarrow \text{Person}(x), \text{Manager}(x) \rightarrow \text{Person}(x) \}$
- Wertebereiche:
 - $O_2 = O_1 \cup \{ \text{kooperiertMit}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y),$
 $\text{berichtetAn}(x, y) \rightarrow \text{Person}(x) \wedge \text{Person}(y) \}$
- Symmetrie:
 - $O_3 = O_2 \cup \{ \text{kooperiertMit}(x, y) \leftrightarrow \text{kooperiertMit}(y, x) \}$
- Transitivität:
 - $O_4 = O_3 \cup \{ \text{berichtetAn}(x, z) \leftarrow \text{berichtetAn}(x, y) \wedge \text{berichtet_an}(y, z) \}$
- Disjunktion:
 - $O_5 = O_4 \cup \{ \text{Manager}(x) \rightarrow \neg \text{Entwickler}(x) \}$

Vereinfacht betrachtet ist eine Ontologie nichts anderes als eine Sammlung derartiger Axiome, die in einer formalen Sprache⁹⁶ – hier in Prädikatenlogik – festgehalten werden.

3.2 Repäsentationsformen von Ontologien

3.2.1 Logik

Die im vorangegangenen Abschnitt erläuterten theoretischen Grundlagen zur Erfassung und Beschreibung von Wissen und Bedeutung mithilfe von Ontologien gilt es für die Anwendung im Case-Based Reasoning zu konkretisieren und in die Praxis umzusetzen. Zunächst sind zur Formalisierung von Wissen Abbildungsmechanismen notwendig die im Folgenden kurz beschrieben werden.

Wie schon in Abschnitt 3.1.5 zur Veranschaulichung des Beispiels vorweggenommen, stellt die formale Logik eine der wichtigsten Methoden zur formalen Beschreibung von Ontologien dar.⁹⁷ Die Prädikatenlogik ist die gebräuchlichste Form der Logik und bildet das Grundgerüst der meisten Ontologiesprachen.

Prädikatenlogische Formeln bestehen aus *Termen*, die Objekte der realen Welt beschreiben. Ein Term kann eine Variable sein oder ein n -stelliges Prädikat (Funktionssymbol), das wiederum Terme als Parameter beinhaltet. Ein Prädikat mit null Stellen ist eine Konstante. Beispiele für die Elemente einer Relation in Anlehnung an das Beispiel aus 3.1.5 sind

- $PN324848$ (Konstante)

95) Vgl. GUARINO/OBERLE/STAAB (2009), S. 11 f.

96) Eine natürlichsprachige Beschreibung ist theoretisch auch möglich, jedoch bietet die „Formalsprachlichkeit“ hinsichtlich der Entwicklung des CBR-Systems Automatisierungspotenzial, welches für Zwecke der KI von entscheidendem Vorteil ist. Vgl. ZELEWSKI (2005), S. 162 f.

97) Vgl. STUCKENSCHMIDT (2009), S. 37.

- *berichtetAn(PN027542,PN21444)* (Anwendung eines zweistelligen Prädikats auf zwei Konstanten)

Atomare Formeln stehen für einzelne Aussagen über den Zustand der Realwelt und bestehen genau aus einem Prädikat, das auf einen Term angewendet wird. Innerhalb einer Formel lassen sich mehrere Prädikate durch verschiedene logische Verknüpfungen zueinander in Bezug setzen, sodass neue Aussagen entstehen. Unter anderem sind dabei folgende Verknüpfungen möglich:

- Negation: \neg
- Konjunktion (logisches „Und“): \wedge
- Disjunktion (logisches „Oder“): \vee
- Implikation (aus A folgt B): \rightarrow
- universelle Quantifizierung (für alle $x_i \in F$): $\forall x_i \in F$
- existenzielle Quantifizierung (es existiert mindestens ein $x_i \in F$): $\exists x_i \in F$

Auf Grundlage dieser syntaktischen Möglichkeiten lassen sich komplizierte Aussagen über die Realwelt tätigen.

3.2.2 Semantische Netze

Eine anschaulichere Darstellung von Beziehungsgeflechten zwischen Objekten als die Prädikatenlogik bieten semantische Netze. Grundlage für diese Darstellungsform bilden gerichtete Graphen.⁹⁸ In einem gerichteten Graph sind Knoten untereinander mit gerichteten Kanten verbunden, die einseitig von einem Start- auf einem Endknoten „zeigen“. Die Kante steht für eine Funktion und hat einen Bezeichner.

In einem semantischen Netz als Darstellung einer Ontologie stehen die Knoten für die Objekte der Realwelt und die Kanten für eine bestimmte Relation zwischen diesen Objekten. Durch die „istEin“-Relationen lassen sich Kategorisierungen der Objekte anschaulich in einem Graph darstellen, wodurch das semantische Netz leicht und intuitiv verständlich wird.⁹⁹ Abbildung 8 zeigt einen beispielhaften Teilausschnitt eines semantischen Netzes in Bezug auf das Software-Unternehmen-Beispiel.

98) Vgl. STUCKENSCHMIDT (2009), S. 29.

99) Vgl. STUCKENSCHMIDT (2009), S. 28.

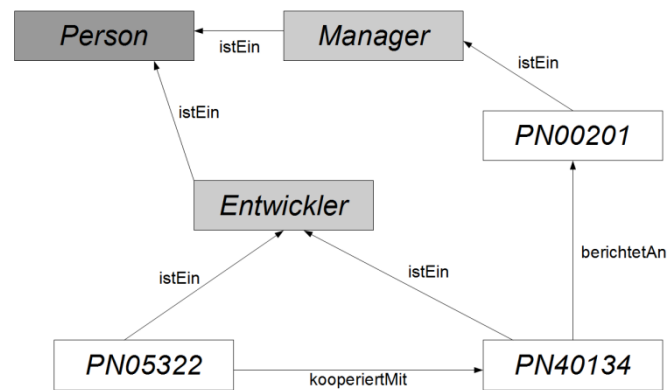


Abbildung 8: Beispiel für ein semantisches Netz¹⁰⁰

Aus diesem Ausschnitt lassen sich bereits einfache Kategorisierungen ableiten – *Manager* und *Entwickler* sind beispielsweise jeweils der Klasse *Person* zuzuordnen.

Für die Repräsentation von Ontologien haben semantische Netze neben der anschaulichen Darstellung den Vorteil, dass sie, basierend auf der Graphentheorie, im Bereich der Künstlichen Intelligenz eine hohe Anzahl an Verarbeitungs-Algorithmen anbietet und die Darstellung problemlos in Computern erfolgen kann.¹⁰¹

Um ein semantisches Netz zu formalisieren, lässt es sich in eine formale Logik, wie beispielsweise die Prädikatenlogik, übersetzen. Eine Möglichkeit hierfür ist es, jede Kante im Netz in eine atomare Formel zu übersetzen. Die Kantenbezeichnungen bilden dann den Namen der Prädikate und die Knoten bilden die Terme. Alle erzeugten atomaren Formeln aus einem semantischen Netz werden dann konjunktiv miteinander verbunden, da jede Formel – bzw. jede Kante – erfüllt sein muss. Eine dementsprechende Übersetzung des semantischen Netzes aus Abbildung 8 in Prädikatenlogik ergäbe folgendes Resultat:

$$\begin{aligned}
 & \text{istEin}(\text{Manager}, \text{Person}) \wedge \text{istEin}(\text{Entwickler}, \text{Person}) \wedge \text{istEin}(\text{PN00201}, \text{Manager}) \\
 & \wedge \text{istEin}(\text{PN40134}, \text{Entwickler}) \wedge \text{kooperiertMit}(\text{PN05322}, \text{PN40134}) \\
 & \text{berichtetAn}(\text{PN40134}, \text{PN00201}) \wedge \text{istEin}(\text{PN05322}, \text{Entwickler})
 \end{aligned}$$

4 Ein Prototyp für ontologiegestütztes Case-based Reasoning in Supply Chains

4.1 Definition der Vorgehensweise für die Entwicklung eines ontologiegestützten Case-based-Reasoning Systems

Zur Planung und Strukturierung des Entwicklungsprozesses richtet sich die Vorgehensweise zur Erstellung des ontologiegestützten CBR-Systems am klassischen Wasserfallmodell aus dem Bereich des Software Engineering.

100) Eigene Darstellung.

101) Vgl. STUCKENSCHMIDT (2009), S. 30.

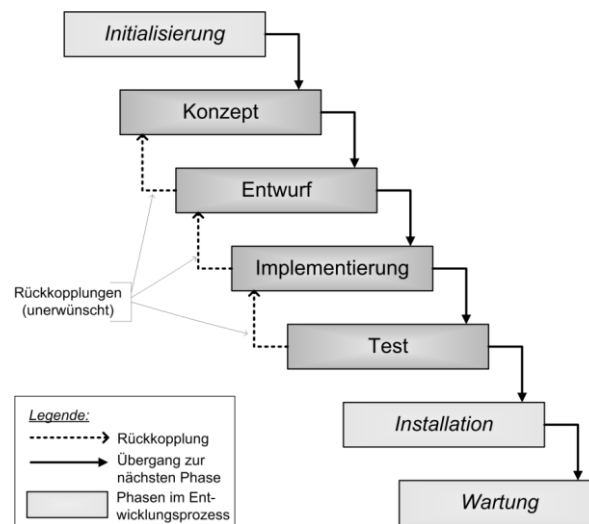


Abbildung 9: Klassisches Wasserfallmodell¹⁰²

Vorteil der Verwendung eines solchen Modells ist die Systematisierung der Herangehensweise an die Entwicklungsaufgabe. Hierdurch werden schon früh im Entwicklungsprozess grundlegende formalisierte Anforderungen und Konzepte zu deren Erfüllung erstellt. Hauptziel dieser Herangehensweise ist die Vermeidung von Fehlerquellen durch übereiltes und unstrukturiertes Software Engineering.

Das in Abbildung 9 gezeigte klassische Wasserfallmodell unterteilt den Entwicklungsprozess von Software in die sieben Phasen Initialisierung, Konzept, Entwurf, Implementierung, Test, Installation und Wartung. Vor allem für komplexere, mehrstufige und parallele Entwicklungsprozesse wurden seit Entwicklung des Wasserfallmodells geeignetere Modelle eingeführt, die aber für den hier beschriebenen flach strukturierten Entwicklungsprozess keinen Mehrwert gegenüber dem Wasserfallmodell bieten würden. Bekannte Beispiele für weiterentwickelte Modelle sind das Spiralmodell, Prototyping und Agile Softwareentwicklung.

Die Untersuchung in dieser Arbeit zielt nicht auf die Erstellung eines fertigen Softwareprodukts ab, sondern will mit der Entwicklung eines Prototyps zeigen, ob und wie eine Verbindung aus ontologischer Wissenspräsentation und Case-Based Reasoning komplexe Entscheidungsprobleme im Supply Chain Management unterstützen kann. Das Ziel in diesem Entwicklungsprozess sollte daher in der Modell-Phase „Test“ durch einen erfolgreichen Test des Prototyps erfolgen, wodurch die Machbarkeit des hier entwickelten CBR-Systems geprüft wird. Daher wird das klassische Wasserfallmodell für die Zwecke dieser Arbeit auf die vier Kernprozesse Konzept, Entwurf, Implementierung und Test reduziert. Rückkopplungen in vorangegangene Phasen sind bei diesen Prozessen möglich, aber nicht wünschenswert. Die einleitende Phase „Initialisierung“, in der ein Projekt gestartet wird, wird bereits durch den Beginn dieser Arbeit abgedeckt.

Die Konzept-Phase besteht aus einer Anforderungsanalyse und einer Systemanalyse¹⁰³. In der Anforderungsanalyse wird spezifiziert, was die zu entwickelnde Software leisten soll. Die Systemanalyse umfasst die Datenmodellierung des CBR-Systems und definiert somit das Fundament für die Software-Architektur.

¹⁰²Eigene Darstellung in Anlehnung an ZÖLLER-GREER (2002), S. 5.

¹⁰³Vgl. ZÖLLER-GREER (2002), S. 9.

In der Entwurfsphase folgt die Planung der Softwarearchitektur. Hierbei werden die einzelnen zu Softwarekomponenten festgelegt, die für die Erstellung der Ontologie sowie die Konstruktion einer Retrieval-Technik für das Case-Based Reasoning verwendet werden.

Der Hauptfokus im hier beschriebenen Entwicklungsprojekt liegt auf der Phase der Implementierung. In dieser Phase wird auf der Grundlage der zuvor erstellten Architekturspezifikation die Software entwickelt. Hauptbestandteile der Entwicklungsarbeiten sind die Erstellung einer Ontologie in einer Ontologiesprache, die Erstellung von Ähnlichkeitsmaßstäben sowie die Erstellung eines Algorithmus für das Retrieval von ähnlichsten Fällen.

Abschließend wird in der Testphase die implementierte Software auf ihre Funktionalität geprüft. Ziel ist es in dieser letzten Phase, die Anwendbarkeit des ontologiegestützten Case-Based Reasoning im Supply Chain Management aufzuzeigen. Hierfür wird die Retrieval-Technik auf eine kleine Falldatenbank mit Testfällen der Praxis angewandt.

Die Vorgehensweise im hier behandelten Entwicklungsprozess wird noch einmal in der folgenden Abbildung veranschaulicht.

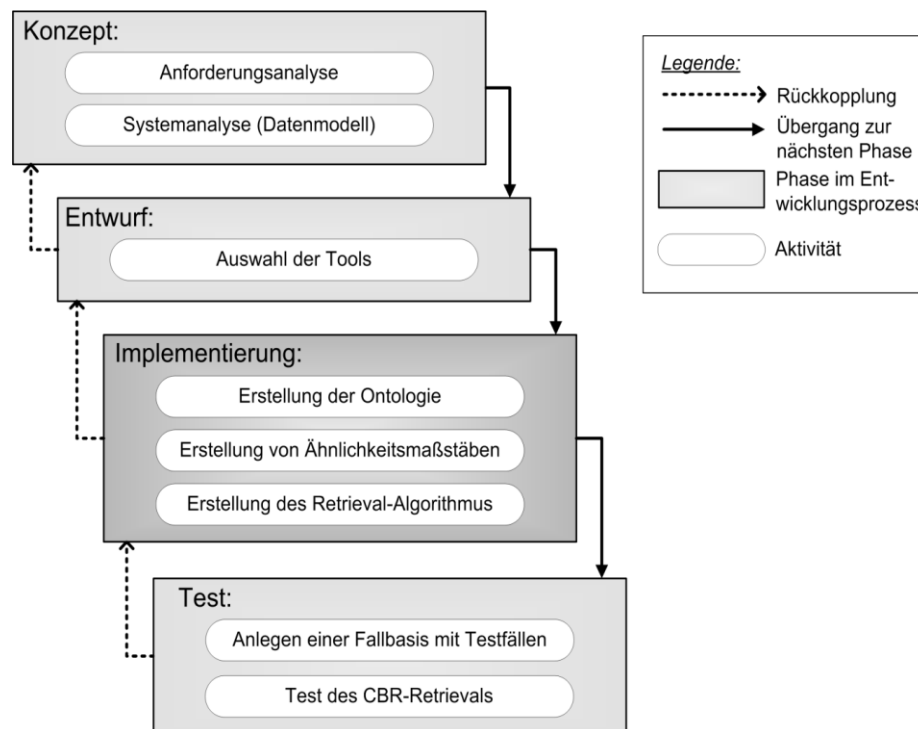


Abbildung 10: Vorgehensweise zur Entwicklung des ontologiegestützten CBR-Prototyps¹⁰⁴

104) Eigene Darstellung.

4.2 Entwicklung des Prototyps

4.2.1 Konzept

4.2.1.1 Anforderungsanalyse

Die Entwicklung eines Prototyps für ontologiegestütztes Case-Based Reasoning unterliegt verschiedenen Anforderungen an die Softwarequalität. Nach dem internationalen Standard *ISO/IEC 9126* sollten folgende Bewertungskriterien beachtet werden, um Softwarequalität sicherzustellen:¹⁰⁵

- Funktionalität
- Zuverlässigkeit
- Benutzbarkeit
- Effizienz
- Instandhaltbarkeit
- Übertragbarkeit

Die erforderlichen Funktionen lassen sich in die Bereiche Ontologie- sowie CBR-Funktionen unterteilen. Die Software soll die Erstellung einer Ontologie mit Ontologiesprachen unterstützen sowie die Ontologie auf Konsistenz überprüfen können. Die CBR-Funktionalitäten, die von der Software abgedeckt werden sollen, bestehen aus der Erstellung von lokalen und globalen Ähnlichkeitsmaßstäben und deren Integration in einen Retrieval-Algorithmus. Grundvoraussetzung für die Erfüllung der CBR-Funktionalität ist der Import der Ontologiebeschreibung.

Die Zuverlässigkeit sollte durch Fehlertoleranz gegeben sein. Systemabstürze und Darstellungsfehler sollten nicht vorkommen. Die Benutzbarkeit soll durch eine intuitiv bedienbare Benutzeroberfläche (GUI) zur Erstellung der Ontologie und der Ähnlichkeitsmaßstäbe gegeben sein. Effizienz und Instandhaltbarkeit spielen zunächst bei der Entwicklung eines Prototyps eine untergeordnete Rolle und fallen erst höher ins Gewicht, wenn die Software zur Marktreife weiterentwickelt wird. Die Übertragbarkeit ist insofern wichtig, als es möglich sein muss, das Ontologiewissen durch Exportieren oder gemeinsame Verwendung für das Case-Based Reasoning nutzbar zu machen.

Zusätzlich wirkt sich die Übertragbarkeit positiv auf die spätere Weiterentwicklung des Prototyps aus, indem die Wiederverwendung des erzeugten Retrieval-Algorithmus (Ontologie, Falldatenbank, Ähnlichkeitsmaßstäbe) ermöglicht wird.

4.2.1.2 Systemanalyse

Das Fundament der späteren Softwarearchitektur wird innerhalb der Systemanalyse errichtet. Hierfür wird ein semantisches Datenmodell erstellt, das die Datenstrukturen und ihre Verflechtungen mit den Softwarekomponenten darstellt.

105) LIGGESMEYER (2009), S. 16 f.

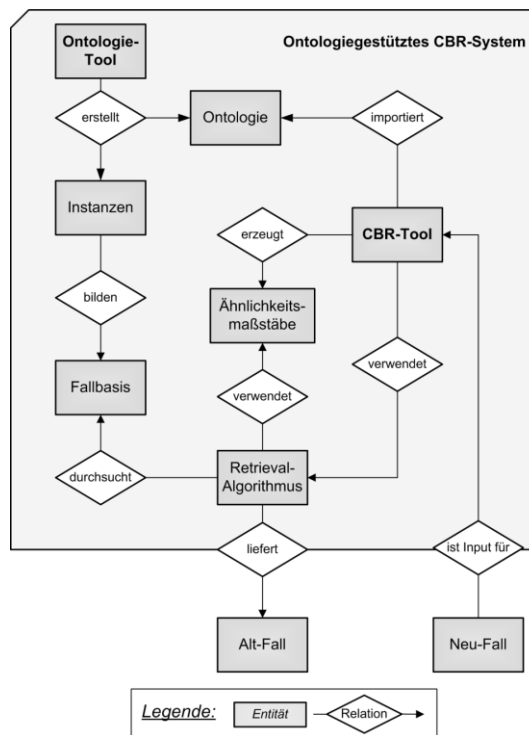


Abbildung 11: Entity-Relationship-Modell¹⁰⁶

Abbildung 11 veranschaulicht die semantische Datenstruktur in Form eines Entity-Relationship-Modells¹⁰⁷. Die in eckigen Kästen dargestellten Entitäten werden durch die raute-förmig dargestellten Relationen miteinander in Verbindung gesetzt, um die inhaltlichen Zusammenhänge der Entitäten im System zu erklären.

Der Prototyp des ontologiegestützten CBR-Systems besteht hauptsächlich aus den beiden Komponenten CBR-Tool und Ontologie-Tool. Das Ontologie-Tool erstellt die Ontologie sowie zugehörige Instanzen der ontologischen Klassen. Das CBR-Tool importiert die Ontologie, um auf deren semantischer Grundlage die Ähnlichkeitsmaßstäbe zu entwickeln. Die Instanzen der ontologischen Klassen bilden als Repräsentation von Realweltobjekten die Falldatenbank des CBR-Systems. Die Ontologie und die zugehörigen Instanzen bilden somit die Schnittstelle zwischen Ontologie-Tool und CBR-Tool.

Bei einer Retrievalanfrage gibt ein Anwender einen Neu-Fall in das System ein, um als Ergebnis mindestens einen ähnlichsten Alt-Fall aus der Falldatenbank zu erhalten. Die Beschreibung des Neu-Falls ist in einer Form gehalten, die konform zur Ontologie ist. An dieser Stelle ergibt sich eine gute Erweiterungsmöglichkeit des hier vorgestellten Systems, indem ein vorgeschalteter Mechanismus aus einem natürlichsprachig beschriebenen Fall die notwendigen Informationen extrahieren könnte, um daraus die formale Fallbeschreibung zu generieren. Um für den nun ins CBR-System eingegebenen Neu-Fall mindestens einen ähnlichsten Alt-Fall zu erhalten, verwendet das CBR-Tool den Retrieval-Algorithmus, um in der Fallbasis nach ähnlichen Fällen zu suchen. Hierfür nutzt der Algorithmus die vom CBR-Tool erstellten globalen Ähnlichkeitsmaßstäbe die die Grundlage für den Vergleich von Fällen darstellen. Die globalen Ähnlichkeitsmaßstäbe sind eine Aggregation gewichteter lokaler Ähnlichkeitsmaßstäbe, die für je ein Attribut eines Falls die Art der Ähnlichkeits-

106) Eigene Darstellung

107) Vgl. BALZERT (2009), S. 199 ff.

bestimmung festlegen. Bei diesen Attributen kann es sich sowohl um quantitativ messbare als auch um qualitativ beurteilbare Werte handeln. Der Algorithmus liefert als Ergebnis der Suche in der Fallbasis eine Liste ähnlichster Alt-Fälle sortiert nach ihren Ähnlichkeitswerten (maximaler Wert = 1) als Output an den anfragenden Anwender des Systems zurück.

4.2.2 Entwurf

4.2.2.1 Auswahl der Softwarekomponenten

Im Rahmen der Entwurfsphase gilt es eine Entscheidung bezüglich der Gestaltung der beiden in der Systemanalyse beschriebenen Komponenten CBR-Tool und Ontologie-Tool zu treffen. Zunächst stellt sich die Frage, ob die Komponenten individuell programmiert werden sollen oder ob auf bereits fertige Software zurückgegriffen werden soll. Die Entscheidung fällt aus Gründen der Zeiterparnis und der Vielfalt verfügbarer Produkte auf die Verwendung fertiger Software, sowohl für die Ontologie-Erstellung als auch für die Komponente, die das Case-Based Reasoning übernimmt. Zusätzlich wird bei der Auswahl der Tools aus Kostengründen auf die Berücksichtigung proprietärer Software verzichtet.

Die marktführenden und frei erhältlichen Tools wurden in einer Dissertation von STEFAN BEIBEL¹⁰⁸ anhand der in Abschnitt 4.2.1.1 aufgeführten Bewertungskriterien Funktionalität, Zuverlässigkeit und Benutzbarkeit bewertet und miteinander verglichen. Bei dieser Untersuchung wurden die Ontologie-Tools Apollo, OntoStudio, OntoTrack, Protégé und Swoop sowie die CBR-Tools CASPIAN, CBR-Shell, Induce-It und MyCBR bewertet. Die Vergleiche wurden mithilfe der Software Expert Choice durchgeführt, die den 1980 vom Mathematiker THOMAS L. SAATY entwickelten Analytischen Hierarchieprozesses (AHP) als Verfahren verwendet. Die AHP-Analyse führte bei den Ontologie-Tools zu folgenden Ergebnissen im relativen Vergleich:

1. **Protégé** **38,0 %**
2. OntoStudio 23,2 %
3. Swoop 21,0 %
4. OntoTrack 11,0 %
5. Apollo 6,7 %

Der deutliche Vergleichssieger Protégé zeichnet sich vor allem im Hauptbewertungskriterium – der Funktionalität „Ontologieerstellung“ – deutlich von der Konkurrenz ab. Ebenso schneidet Protégé im Punkt Benutzbarkeit der GUI eindeutig besser ab als alle Konkurrenten.

Im Vergleich der CBR-Tools fiel das Ergebnis etwas knapper aus:

1. **MyCBR** **40,1 %**
2. Induce-It 35,7 %
3. CBR Shell 17,4 %
4. CASPIAN 6,8 %

108) Vgl. BEIBEL (2011), S. 85 ff.

MyCBR punktet hierbei vor allem mit einer guten Übertragbarkeit durch den Import von Ontologiewissen sowie der besten Benutzbarkeit im Vergleich zur Konkurrenz.

Auf der Grundlage der Resultate der oben beschriebenen Untersuchung werden für den Entwurf des ontologiegestützten CBR-Tools für Supply Chains die Tools Protégé und MyCBR verwendet. Ein zusätzlicher Vorteil bei dieser Entscheidung ist, dass MyCBR neben einer Standalone-Version auch als Plug-in für Protégé erhältlich ist und somit von einer vereinfachten Integration der beiden Komponenten auszugehen ist.

4.2.2.2 Protégé

Protégé¹⁰⁹ ist ein Editor für Ontologien und wurde vom STANFORD CENTER FOR BIOMEDICAL INFORMATICS RESEARCH entwickelt. Ursprünglich wurde die Software konzipiert, um für medizinische Forschungszwecke Ontologien zu erstellen. Das Programm ist in der Programmiersprache Java geschrieben und ist dadurch plattformunabhängig zu betreiben. Aktuell unterliegt die Software der General Public License (GPL) und ist somit inklusive Quellcode frei verfügbar. Mit Protégé ist es auf zwei unterschiedliche Arten möglich, Ontologien zu modellieren. Zum einen existiert ein „frame-basierter“ Ansatz der Wissensrepräsentation, bei dem Informationen über ein bestimmtes Domänenwissen in einer hierarchischen Struktur von Klassen, „Slots“ (zu Klassen assoziierte Attribute und Relationen) und Instanzen von Klassen (Individuen) dargestellt werden. Zum anderen existiert der Ansatz Protégé-OWL, der auf einem Standard aus dem Semantic Web, der Web Ontology Language (OWL), basiert. Im zweiten Ansatz ist es möglich, Ontologien mit OWL oder RDF (Resource Description Framework) zu erstellen. Zusätzlich zum frame-basierten Ansatz ist es hierbei möglich, durch logische Mechanismen implizites Wissen aus bestehenden Datenstrukturen zu schlussfolgern.

Protégé ermöglicht über Benutzerschnittstellen die Konstruktion und Speicherung von Ontologien. Zusätzlich werden automatisch Formulare aus den Klassenbeschreibungen generiert, die eine direkte Eingabe von Instanzen ermöglichen. Diese Formulare sind durch den Anwender anpassbar.

Über die Java-basierten Schnittstellen (API's) ist es möglich, externe Software an Protégé anzubinden und somit die erstellten Ontologien zu nutzen. Die erzeugten Klassenstrukturen der Ontologie können als Java-Code exportiert werden. Durch die Plugin-Struktur können Software-Komponenten auch in Protégé integriert werden – wie es bei MyCBR der Fall ist. Protégé ist kompatibel zum Open-Knowledge-Base-Connectivity-Protokoll.¹¹⁰

4.2.2.3 MyCBR

MyCBR¹¹¹ wurde vom DEUTSCHEN FORSCHUNGSZENTRUM FÜR KÜNSTLICHE INTELLIGENZ in Kaiserslautern entwickelt und unterliegt ebenso wie Protégé der GPL. Ziel der Entwicklungsarbeiten ist

109) Vgl. BMIR (2011) o. S.

110) Das Open Knowledge-Base-Connectivity-Protokoll (OKBC) bietet den standardisierten Zugriff auf Wissen verschiedener Repräsentationssprachen. OKBC definiert generische Operationen für den Zugriff auf in Applikationen gespeichertes Wissen. Vgl. GAŠEVIĆ/DJURIĆ/DEVEDŽIĆ (2006), S. 39 ff.

111) Vgl. DFKI (2011) o. S.

es, die Funktionalität von Case-based Reasoning zu integrieren und gleichzeitig einfache Handhabung, Erweiterbarkeit, Adaptierbarkeit und schnelles Prototyping zu gewährleisten. MyCBR erweitert Protégé um eine alternative Instanzerzeugung mittels CSV-Import.

Mit MyCBR ist es möglich, lokale Ähnlichkeitsmaßstäbe für Attribute von Klassen sowie globale Ähnlichkeitsmaßstäbe für Klassen zu erstellen. Der Retrieval-Mechanismus benutzt die Ähnlichkeitsmaßstäbe, um Fälle anhand ihrer Attribute und Klassen miteinander zu vergleichen. Die Ähnlichkeitsmaßstäbe die mit MyCBR erstellt wurden, können in Form von XML-Dateien exportiert werden. Je nach Typ des Attributs bietet MyCBR unterschiedliche Formen der Ähnlichkeitsbestimmung an:

- Taxonomie,
- Tabelle,
- Abstand,
- Textvergleich bei Strings und
- geordnete Werte.

Wenn die vorgegebenen und anpassbaren Formen für Ähnlichkeitsbestimmungen nicht ausreichen, kann entweder ein externes Java-Programm angebunden werden oder innerhalb von MyCBR ein Skript erstellt werden, das in der Programmiersprache Python¹¹² geschrieben werden muss. Durch diese Mechanismen lassen sich gegebenenfalls auftretende Lücken in den Funktionalitäten durch eigene Programmbausteine umgehen.

4.2.2.4 Integration der beiden Komponenten

Für die Entwicklung des ontologiegestützten CBR-Systems ist es notwendig, die beiden Komponenten Ontologie-Tool (Protégé) und CBR-Tool (MyCBR) miteinander zu verknüpfen, damit das Ontologiewissen – wie in Abbildung 11 dargestellt – das Fundament für die Falldatenbank und die Ähnlichkeitsmaßstäbe des Case-based Reasonings bilden kann. Da MyCBR auf der Basis von Protégé entwickelt wurde und sich als Plugin integrieren lässt, wird die Definition von Schnittstellen bereits vorweggenommen und muss nicht zusätzlich konfiguriert werden.

Bei der Installation der Systeme ist darauf zu achten, dass MyCBR nur kompatibel zu den Protégé-Versionen 3.x ist. Im Rahmen dieser Arbeit wurde die Version 3.4.4 verwendet. Die verwendete MyCBR-Version lautet 2.6.6. Des Weiteren ist es wichtig, dass in Protégé als Modellierungssprache der interne Standard „Protégé Files“ ausgewählt wird, bei dem die Instanzen in „.pins“-Dateien und die Definitionen der Klassen und zugehöriger Slots in „.pont“-Dateien gespeichert werden.¹¹³

112) Es handelt sich hierbei genauer gesagt um Jython, eine Java-Implementierung von Python, die es ermöglicht, in Python geschriebenen Code innerhalb der Java-Plattform auszuführen.

113) Während des Entwicklungsprozesses kam es zu einer Rückkopplung von der Phase „Implementierung“ zur „Entwurf“, da zunächst eine Ontologie im OWL-Format erstellt wurde, aber bei dem Versuch, diese Ontologie als Grundlage für MyCBR zu benutzen Systemabstürze eintraten. Die Konvertierung der OWL-Ontologie zu Protégé-Files schlug fehl, auch hier kam es zu Systemabstürzen (von MyCBR verursachte Java-Exceptions). Die Phase Implementierung musste daher wiederholt werden.

Die Schnittstelle der beiden Komponenten liegt in den beiden Protégé-Files (Pins und Pont). MyCBR importiert daraus die Instanzen (Pins) und die Metadaten aus deren Klassenbeschreibungen (Pont). Aufbauend auf den importierten Daten werden Ähnlichkeitsmaßstäbe entwickelt und getrennt in XML-Dateien abgelegt.

4.2.3 Implementierung

4.2.3.1 Umfang der Implementierung

Im Rahmen der Implementierung wird das Konzept aus Abschnitt 4.2.1.2 auf Grundlage des Entwurfs im vorangegangenen Abschnitt zu einem lauffähigen ontologiegestützten CBR-System entwickelt. Grundlage für diesen Prototyp ist zunächst die Entwicklung einer Ontologie aus der Domäne Supply Chain Management. Darauf aufbauend werden für die Klassen und Attribute (Slots) der Ontologie lokale Ähnlichkeitsmerkmale erstellt, die anschließend zu globalen Ähnlichkeitsmaßstäben aggregiert werden und in dieser Form Gegenstand des Retrieval-Algorithmus werden.

4.2.3.2 Erstellung der Ontologie

4.2.3.2.1 Vorgehen zur Ontologierstellung

Im Gegensatz zur Softwareentwicklung existieren zur Ontologierstellung noch keine allgemein akzeptierten Vorgehensmodelle.¹¹⁴ Es sind aber einige Ansätze vorhanden¹¹⁵, wie beispielsweise die „Cyc-Methode“, die „Methontology“ sowie der „KACTUS-Ansatz“. Die hier verwendete Vorgehensweise ist angelehnt an die Methode von NOY & MCGUINNESS¹¹⁶, die aus sieben Entwicklungsschritten besteht:

1. Beschreibung der Ontologiedomäne,
2. Wiederverwendung bestehender Ontologien,
3. Identifikation relevanter Begriffe,
4. Festlegung der Klassen und der Klassenhierarchie,
5. Definition der Slots (Attribute und Relationen),
6. Definition der Slot-Eigenschaften und
7. Erstellen von Instanzen.

Die Ontologierstellung wird im Folgenden mithilfe der ersten sechs Schritte gegliedert. Die Erstellung von Instanzen zum Aufbau einer Falldatenbank für das Case-based Reasoning wird inhaltlich

114) Vgl. STUCKENSCHMIDT (2005), S. 160.

115) Vgl. GOMEZ-PEREZ/FERNANDEZ-LOPEZ/CORCHO-GARCIA (2003), S. 113 ff.

116) Vgl. NOY/MCGUINNESS (2001), S. 4 ff.

der Testphase des CBR-Systems zugeordnet (siehe Abbildung 10) und erfolgt daher erst in Abschnitt 4.2.4.1.¹¹⁷

4.2.3.2.2 Beschreibung der Ontologieerstellung

Die Ontologiedomäne als Anwendungsgebiet für das Case-Based Reasoning ergibt sich mit dem Supply Chain Management als Teil der betriebswirtschaftlichen Disziplin der Logistik. Es gilt vor der eigentlichen Erstellung der Ontologie den betrachteten Ausschnitt der Domäne abzugrenzen, indem festgelegt wird, welche Aspekte berücksichtigt und welche Aspekte nicht berücksichtigt werden sollen. Diese Abgrenzung ist von entscheidender Notwendigkeit. Es sollte ein optimaler Detaillierungsgrad erreicht werden, der zum einen den Aufwand für die Ontologieerstellung begrenzt, und zum anderen eine aussagekräftige Konzeptualisierung der Realwelt zulässt.

Ein bewährtes Instrument zur Fokussierung einer Ontologie auf relevante Aspekte eines speziellen Anwendungszwecks sind die sogenannten Competency Questions¹¹⁸. Hierbei handelt es sich um Fragen, deren Beantwortung in der Kompetenz der zu entwickelnden Ontologie liegen sollte. Die Fragen dienen neben der Fokussierung der Domäne später zum Test, ob die erstellte Ontologie genug Informationen bietet, um diese zu beantworten, und ob der gewählte Detaillierungsgrad ausreichend ist, um die betrachtete Anwendungsdomäne zum repräsentieren.

Die Competency Questions sind als Ausgangspunkt für die Erstellung einer Ontologie anzusehen und sind während der weiteren Entwicklung weiterhin anpassbar und erweiterbar. Die Fragen zur Ontologie im Supply Chain Management lauten zunächst:

- Wie ist eine Supply Chain, die von einem Kaufvertrag ausgelöst wurde, aufgebaut?
- Welche Eigenschaften hat ein einzelner Versandprozess?
- Welche Akteure wirken bei der Gestaltung und Durchführung von Supply Chains mit?
- Welche Faktoren beeinflussen die Entscheidung zur Beauftragung von Logistikdienstleistern?
- Welche Faktoren beeinflussen die Entscheidung zur Wahl der Route und der genutzten Verkehrsträger?
- Wie lassen sich die verschiedensten Arten von Gütern oder Sachgütern unterscheiden und kategorisieren?

Die Aspekte, die in der Ontologieerstellung berücksichtigt werden, zielen auf die abstrakte Abbildung von globalen Transportfällen, die von einer Vertragsbeziehung zwischen einem Verkäufer und einem Käufer ausgelöst werden. Hierbei werden die aufeinanderfolgenden Transportabschnitte einer Supply Chain auf teils unterschiedlichen Verkehrsträgern betrachtet und in Verbindung mit den

117) Diese Vorgehensweise hat sich während der Ontologieerstellung mit Protégé und integriertem MyCBR-Plugin als unausweichlich herausgestellt, da mit bereits vorhandenen Instanzen und nachträglichen Änderungen an deren Klassenbeschreibungen mehrfach Systemabstürze hervorgerufen wurden.

118) Vgl. STUCKENSCHMIDT (2009), S. 161 f. und NOY/MCGUINNESS (2001), S. 4.

organisierenden, prüfenden und durchführenden Akteuren gebracht. Verschiedene Faktoren, die die Gestaltung solcher Supply Chains beeinflussen, werden in die Ontologierstellung einfließen.

Wie bereits in Abschnitt 1.2 angesprochen, eignet sich der Einsatz von Case-Based Reasoning insbesondere in Anwendungsgebieten, die zu komplex sind, um deren Probleme mit einfacher Befolgung von Regeln oder Algorithmen zu bewältigen. Der hier betrachtete Bereich der „Logistik-Welt“ zeigt ein zwar stark abstrahiertes, aber dennoch ausreichend komplexes Einsatzgebiet und eignet sich daher ideal, um die Machbarkeit eines ontologiegestützten CBR-Systems zur Unterstützung betriebswirtschaftlicher Entscheidungsprozesse zu untersuchen.

4.2.3.2.3 Wiederverwendung bestehender Ontologien

Um wiederverwendbare Ansätze für die Ontologie der beschriebenen Anwendungsdomäne zu finden, wurde auf diversen Internetplattformen recherchiert¹¹⁹. Die Suchergebnisse überschneiden sich teilweise mit der Thematik. Es lies sich aber keine Ontologie finden, die den konkreten Anforderungen gerecht wurde. Zusätzlich sind die in der Online-Recherche gefundenen Ergebnisse hauptsächlich in OWL erstellten Ontologien und somit nicht problemlos für die hier geschilderten Zwecke zu übernehmen.

Aus den genannten Gründen und der Abschätzung, dass selbst bei wiederverwendbaren Teilen von bestehenden Ontologien das Risiko besteht, dass der Aufwand für Anpassung, Integration und Validierung der bestehenden Bausteine den Mehrwert der Wiederverwendung schmälern kann, wurde im Weiteren auf die Wiederverwendung bestehender Ontologien verzichtet.

4.2.3.2.4 Identifikation relevanter Begriffe

Die Ontologie beruht, wie in den Abschnitten 3.1.4 und 3.1.5 erläutert, auf „spezifizierten Konzeptualisierungen“. Die in einer Ontologie enthaltenen Klassen, Relationen und Instanzen gilt es nun in Form von Begriffen der Anwendungsdomäne zu identifizieren und in den folgenden Schritten zu spezifizieren.

Um den Realweltausschnitt der Logistikprozesse in Supply Chains zu ergründen werden zwei Arten von Wissensquellen verwendet. Der Kern der Wissensschöpfung hat im Rahmen von informellen Interviews mit einem Experten aus der Logistikbranche stattgefunden¹²⁰. Zusätzlich zu den in den Interviews gewonnen Informationen wurde in der Fachliteratur und frei verfügbaren Internetressourcen nach verwertbaren Informationen recherchiert. Die aus den Interviews mit dem Logistikexperten hervorgegangene Stichwortsammlung ist in der folgenden Tabelle alphabetisch sortiert aufgeführt.

119)Die Online-Recherche wurde durchgeführt im DAML Ontologieverzeichnis (URL: „<http://www.ai.sri.com/daml/ontologies/>“) sowie unter der Semantic Web Suchmaschine Swoogle („<http://swoogle.umbc.edu/>“), wo mit u. a. folgenden Suchbegriffen gesucht wurde: „Supply Chain“, „Logistic“, „Transport“, „Distribution“, „Shipment“.

120)Im Folgenden wird der Begriff „Logistikexperte“ für Herrn HORST LAUTENSCHLÄGER verwendet. Herr Lautenschläger ist Geschäftsführer der relamedia GmbH in Herne und hat sich dazu bereit erklärt, diese Arbeit beratend zu unterstützen.

Akkreditiv	Lieferbedingungen	Termin
Akteur	Luftverkehr	Transportweg
Binnenschifffahrt	Massenleistungsfähigkeit	Unternehmen
Eisenbahnverkehr	Menge	Verkäufer
Empfänger	Ort	Verpackung
Flexibilität	Person	Versender
Frachtführer	Pünktlichkeit	Wareabmessungen
Gefahrgut	Regelmäßigkeit	Warenart
Käufer	Schnelligkeit	Warengewicht
Käuferbeziehung	Seeschifffahrt	Ware
Kosten	Sicherheit	Warenwert
Lieferant	Spediteur	Zahlungsabwicklung
Lieferantenbeziehung	Straßenverkehr	Zoll-Agent

Tabelle 1: Relevante Begriffe für die Ontologierstellung

4.2.3.2.5 Festlegung der Klassen und der Klassenhierarchie

Die Abbildung der in Tabelle 1 aufgeführten Begriffe in ein geeignetes formales Modell ist der wesentliche Schritt bei der Erstellung der Ontologie¹²¹. Es gilt zu entscheiden, welche der Begriffe als Klassen dargestellt werden und welche Begriffe in Form einer Relation zwischen Klassen abgebildet werden sollen. Manche der Begriffe fließen vorerst nicht in die Klassenhierarchie ein, da sie als Attribute oder Relationen abgebildet werden, die in der Darstellung der Klassenhierarchie zunächst noch ausgeblendet werden.

121) Vgl. STUCKENSCHMIDT (2009), S. 166.

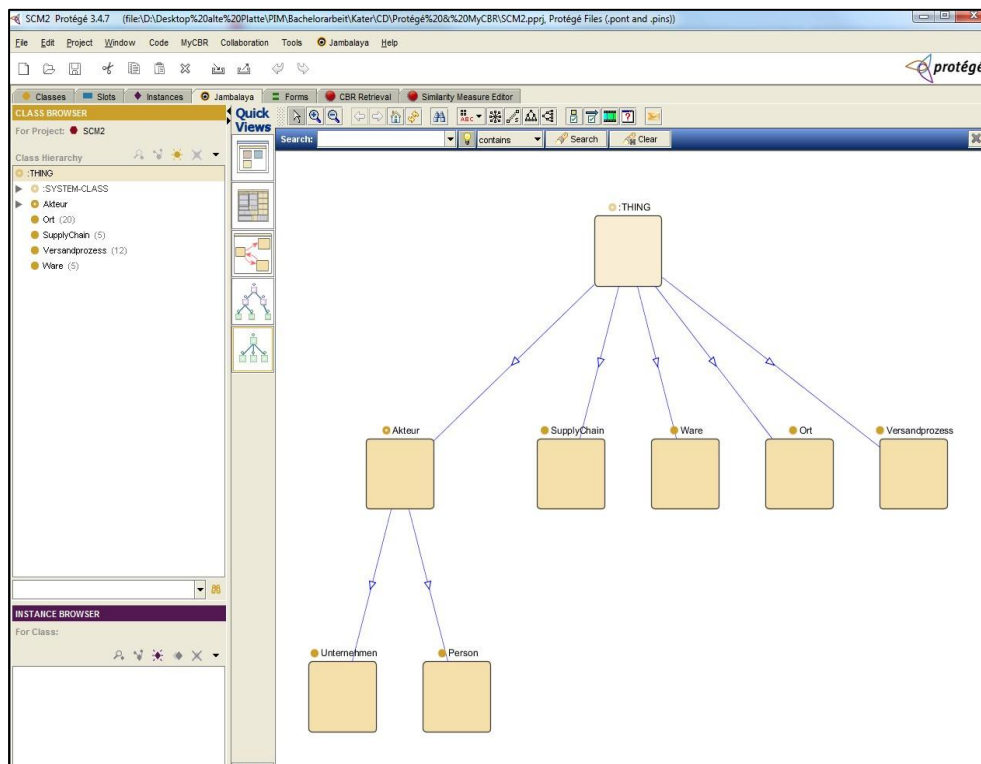


Abbildung 12: In Protégé dargestellte Klassenhierarchie

Aus der Menge der Begriffe wurde nur eine Minderheit für die Abbildung in einer Klasse ausgewählt. Hierbei handelt es sich um folgende Klassen:

- Supply Chain
- Versandprozess
- Akteur
- Unternehmen
- Person
- Ware
- Ort

Die Klassen werden in Protégé immer als Unterklasse einer bereits bestehenden Klasse angelegt. Die auf der Abstraktionsebene am höchsten gelegene Klasse ist „Thing“, von ihr werden alle Klassen einer Ontologie zusammengefasst. Abbildung 12 zeigt einen Screenshot aus Protégé mit einer Darstellung der Klassenhierarchie. Das Diagramm wurde automatisch mithilfe des Plugins Jambalaya generiert.

Die Klassen Unternehmen und Person sind Unterklassen der Klasse Akteur. Grund für die flache Struktur der Klassenhierarchie ist die Tatsache, dass der Hauptteil der Begriffe aus Tabelle 1 sich eher eignet, als Attribut einer Klasse oder als Relation zwischen Klassen abgebildet zu werden. Beispielsweise übernehmen die bei Versandprozessen beteiligten Akteure verschiedene Rollen, wie z.B. Spediteur, Frachtführer, Versender oder Empfänger. Jedoch lässt sich keiner der Akteure strikt als Akteur einer Klasse (z. B. die Klasse Versender) beschreiben, er kann auch gleichzeitig in einem anderen Fall Empfänger, Frachtführer oder ein sonstiger Rollenträger sein. Die ausgeführte Rolle

im Versandprozess kann von Fall zu Fall unterschiedlich sein und lässt sich in der Ontologie geschickter als Relation zwischen Akteur und Versandprozess darstellen.

Die Klasse SupplyChain repräsentiert den Ausgangspunkt für einen Fall im Case-based Reasoning, d.h. beim Vergleich zweier Fälle werden im CBR-System Instanzen der Klasse SupplyChain miteinander verglichen, die je nach Ausprägung mit Instanzen anderer Klassen verknüpft sind.

4.2.3.2.6 Definition der Slots (Attribute und Relationen)

Allein mit der Festlegung der Klassen und der Klassenhierarchie stehen noch nicht genügend Informationen zur Verfügung, um die in Abschnitt 4.2.3.2.2 aufgeführten Competency Questions beantworten zu können. Daher geht es im nächsten Schritt darum, die zuvor erstellten Klassen genauer zu beschreiben. Für die Beschreibung der Attribute von Klassen stehen in der Ontologiesyntax der Protégé-Files sogenannte Slots zur Verfügung.

Es gibt drei verschiedene Arten von Attributen, die als Slot einer Klasse zugeordnet werden können:¹²²

- Intrinsische Attribute (Eigenschaften, die zu einer Klasse selbst gehören und Instanzen der Klasse zu dem machen was sie sind),
- Extrinsische Attribute (Ausprägungen, die der Klasse verliehen wurden und änderbar sind),
- Beziehungen (Relationen) zu anderen Klassen.

Die Attribute werden zunächst den Klassen auf der untersten Hierarchieebene zugeordnet. Anschließend werden diese Attribute generalisiert. Generalisierung findet statt, wenn Attribute, die in allen Unterklassen einer Klasse vorhanden sind, der übergeordneten Klasse zugeordnet werden. Alle Unterklassen der übergeordneten Klasse „erben“ dann diese Attribute. Die folgende Aufzählung zeigt eine Übersicht der Klassen und zugehöriger Slots (geerbte Attribute werden nicht aufgeführt). Bei der Schreibweise ist zu beachten, dass laut der im Protégé-Handbuch beschriebenen Konvention Attributs- und Relationsnamen immer mit einem Kleinbuchstaben beginnen sollten, weitere Wörter in der Bezeichnung werden großgeschrieben.¹²³

- Akteur:
 - Relationen: *istLieferantFür, ort*
 - Attribut: *name*
- Person:
 - Relation: *istEin (Akteur)*
- Unternehmen:
 - Relation: *istEin (Akteur)*
 - Attribut: *unternehmensgröße*

122) Vgl. NOY/MCGUINNESS (2001), S. 4 ff.

123) Vgl. HORRIDGE (2011), S. 26.

- Ort:
 - Attribute: *handelszone, ortsname, postcode, weltregion*

- SupplyChain:
 - Relationen: *befördertWare, beginntIn, endetIn, hatKäufer, hatVerkäufer*
 - Attribute: *beziehungVerkäuferZuKäufer, flexibilität, kosten, beziehungVerkäuferZuLieferant, lieferbedingungen, massenleistungsfähigkeit, pünktlichkeit, regelmäßigkeit, schnelligkeit, sicherheit, zeitfrist, zahlungsabwicklung*

- Versandprozess:
 - Relationen: *beginntIn, endetIn, hatEmpfänger, hatFrachtführer, hatSpediteur, hatVersender, hatZollagent, istTeilVon*
 - Attribut: *nutztVerkehrsträger*

- Ware:
 - Attribute: *menge, breite, länge, höhe, gesamtgewicht, gesamtwert, taricNr, warenart, warenbezeichnung, gefahrgutklasse*

4.2.3.2.7 Definition der Slot-Eigenschaften

Slots lassen sich durch verschiedenartige Eigenschaften weiter spezifizieren.¹²⁴ Grundlegend ist zunächst die Zuweisung einer Domain. Die Domain ist die Menge von Klassen, die den betreffenden Slot beinhalten. Die Domain der Relation *beginntIn* beinhaltet beispielsweise die Klassen *SupplyChain* und *Versandprozess*. Das zweite Spezifikationsmerkmal von Slots ist die Kardinalität. Durch die Kardinalität wird definiert, wie viele Werte ein Slot annehmen kann. In Protégé kann bei der Festlegung der Kardinalität ein Minimum und ein Maximum für die Anzahl der möglichen Slotwerte ausgewählt werden.¹²⁵

Ein weiteres wichtiges Spezifikationsmerkmal von Slots ist der Wertetyp. Protégé bietet acht verschiedene Wertetypen an, von denen die folgenden fünf im hier beschriebenen Entwicklungsprozess verwendet werden:

- String (einfache Zeichenkette),
- Integer (ganzzahlige Werte),

¹²⁴) Vgl. NOY/MCGUINNESS (2001), S. 10 ff.

¹²⁵) Der Spielraum bei der Konfiguration der Kardinalität wurde im Entwicklungsprozess leider seitens MyCBR eingeschränkt. MyCBR erlaubt keine multiplen Kardinalitäten bei Relationen. Ursprünglich war angedacht, mehrere Vorlieferanten von Einzelteilen dem Haupttransport vorzuschalten. Es ist aber z.B. nicht möglich, einer Ware eine Relation „bestehtAus“ zuzuordnen, die auf mindestens zwei Instanzen einer anderen Klasse verweist. Auf eine Anfrage an das Entwicklerteam von MyCBR kam als Antwort der Verweis auf die neuere Version 3.0, die allerdings nicht mit Protégé kompatibel ist.

- Float (Gleitkommazahlen),
- Symbol,
- Instance und
- Class.

Neben den primitiven Datentypen String, Integer und Float werden noch die Typen Symbol, Instance und Class verwendet. Der Wertetyp Instance bezieht sich ausschließlich auf Relationen und beinhaltet die sogenannte Range, die bestimmt, welche Klassen in der betreffenden Relation zu den Klassen aus der Domain stehen. Im Beispiel der Relation *beginntIn* ist dies die Klasse *Ort*.

Der Wertetyp Symbol beinhaltet eine vom Protégé-Benutzer vordefinierte Auswahl an erlaubten Werten. Z.B. enthält der Slot *beziehungVerkäuferZuKäufer* die Auswahl der Begriffe „selbes Unternehmen“, „getrenntes Unternehmen“ und „verbundenes Unternehmen“. Der Wertetyp Class wird bei der Relation *istEin* verwendet, um eine Klasse einer übergeordneten Klasse zuzuordnen.

Tabelle 2 und Tabelle 3 zeigen eine Übersicht der verwendeten Slots und ihrer Eigenschaften.

Klasse	Slot			
	Name	Typ	Werttyp	Kardinalität
Akteur	name	Attribut	String	1
	istLieferantFür	Relation	Instance	0-1
	ort	Relation	Instance	1
Person	istEin	Relation	Instance	1
Unternehmen	unternehmensgröße	Attribut	Symbol	1
	istEin	Relation	Instance	1
Versandprozess	nutztVerkehrsträger	Attribut	Symbol	1
	beginntIn	Relation	Instance	1
	endetIn	Relation	Instance	1
	hatEmpfänger	Relation	Instance	1
	hatFrachtführer	Relation	Instance	1
	hatSpediteur	Relation	Instance	0-1
	hatVersender	Relation	Instance	1
	hatZollagent	Relation	Instance	0-1
	istTeilVon	Relation	Instance	1
Ware	menge	Attribut	Float	1
	breite	Attribut	Float	1
	höhe	Attribut	Float	1
	länge	Attribut	Float	1
	gesamtwicht	Attribut	Float	1
	gesamtwert	Attribut	Float	1
	taricNr	Attribut	String	1
	warenart	Attribut	Symbol	1
	warenbezeichnung	Attribut	String	0-1
Ort	handelszone	Attribut	Symbol	0-1
	ortsname	Attribut	String	1
	postcode	Attribut	String	0-1
	weltregion	Attribut	Symbol	1

Tabelle 2: Übersicht der verwendeten Slots (Teil 1)

Klasse	Slot			
	Name	Typ	Wertetyp	Kardinalität
SupplyChain	beziehungVerkäuferZuKäufer	Attribut	Symbol	1
	beziehungVerkäuferZuLieferant	Attribut	Symbol	0-1
	flexibilität	Attribut	Symbol	1
	kosten	Attribut	Symbol	1
	lieferbedingungen	Attribut	Symbol	1
	massenleistungsfähigkeit	Attribut	Symbol	1
	pünktlichkeit	Attribut	Symbol	1
	regelmäßigkeit	Attribut	Symbol	1
	schnelligkeit	Attribut	Symbol	1
	sicherheit	Attribut	Symbol	1
	zeitfrist	Attribut	Integer	0-1
	befördertWare	Relation	Instance	1
	beginntIn	Relation	Instance	1
	endetIn	Relation	Instance	1
	hatKäufer	Relation	Instance	1
	hatVerkäufer	Relation	Instance	1
	zahlungsabwicklung	Attribut	Symbol	0-1

Tabelle 3: Übersicht der verwendeten Slots (Teil 2)

Wertetyp „Instance“:

Die Slots mit dem Wertetyp Instance bilden die Relationen zwischen den Klassen der Ontologie und erweitern damit die in Abschnitt 4.2.3.2.5 erläuterte Klassenhierarchie um ein dichteres Beziehungsgeflecht. Der Slot *ort* bildet eine Relation von der Klasse Akteur zur Klasse Ort. Der Slot wird an die Subklassen Person und Unternehmen vererbt und gibt damit einen Wohnsitz für eine Person bzw. einen Firmensitz für ein Unternehmen an.

Die Klasse Versandprozess beinhaltet mit den Slots *beginntIn* und *endetIn* zwei Relationen zur Klasse Ort, der Start- und den Endpunkt eines Transportabschnitts markieren. Des Weiteren gibt es fünf Relationen, die die Klasse Versandprozess mit der Klasse Akteur verknüpfen (*hatEmpfänger*, *hatFrachtführer*, *hatSpediteur*, *hatVersender*, *hatZollagent*). Durch diese Relationen wird das Konzept der Rolle repräsentiert (siehe auch Ende Abschnitt 4.2.3.2.5). Wenn ein Akteur beispielsweise in einem Versandprozess die Rolle des Frachtführers übernimmt, so hat die Instanz dieses Versandprozesses die Verknüpfung zur Akteur-Instanz durch den Slot *hatFrachtführer*. Eine weitere Relation (*istLieferantFür*), die eine Rolle repräsentiert, geht von einem Akteur in der Lieferantenrolle aus, und verweist auf die Klasse SupplyChain.¹²⁶ Der Slot *istTeilVon* ordnet den einzelnen Versandpro-

126) Die Relation *istLieferantFür* ist anders als die anderen Relationen, die Rollen darstellen, entgegengesetzt gerichtet, d.h. sie zeigt nicht auf den Akteur, sondern vom Akteur auf die Klasse SupplyChain. Dies wurde hier so gehand-

zess zur übergeordneten Supply Chain zu. Mehrere Instanzen der Klasse Versandprozess können durch diesen Slot auf dieselbe Instanz der Klasse SupplyChain verweisen, wodurch sich die Supply Chain als Kette von beliebig vielen Versandprozessen darstellen lässt. Die Reihenfolge der Versandprozesse in einer Supply Chain wird durch die Übereinstimmung von Endpunkt des Vorgängers (*endetIn*) mit dem Startpunkt des Nachfolgers (*beginntIn*) fixiert. Auf diese Weise lassen sich kleinere Änderungen im Transportablauf, wie beispielsweise der Frachtführerwechsel beim Zusammenkoppeln verschiedener Güterwaggons, darstellen, indem der Ort des Zusammenkoppeln den Knotenpunkt zwischen zwei Versandprozessen markiert.

Die Klasse SupplyChain besitzt ebenso wie die Klasse Versandprozess zwei Slots, die auf je einen Akteur als Rolleninhaber verweisen. Durch die Slots *hatKäufer* und *hatVerkäufer* werden die Vertragspartner des (der Supply Chain zugrundeliegenden) Kaufvertrags gekennzeichnet. Ein weiterer Slot (*befördertWare*) bringt die Supply Chain mit der Klasse Ware in Beziehung. Die Slots *beginntIn* und *endetIn* sind analog zum Versandprozess und markieren in der Klasse SupplyChain den Startpunkt (z.B. der Ort des Lieferanten oder des Verkäufers) sowie den Endpunkt einer gesamten Supply Chain (z.B. Ort des Käufers).

In Abbildung 13 wird zusammenfassend das gesamte Beziehungsgeflecht der Ontologie dargestellt.

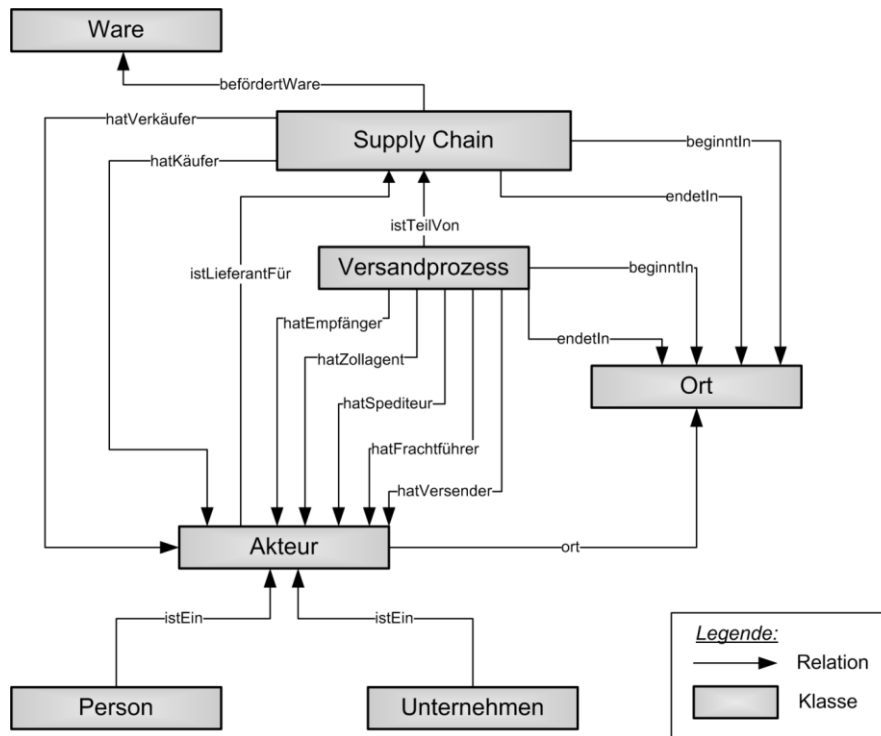


Abbildung 13: Relationen innerhalb der Ontologie

Wertetyp „Symbol“:

Einige der Slots besitzen den Wertetyp Symbol und bekommen feste Werte zugeordnet, die beim Anlegen von Instanzen ausgewählt werden können. Der Slot *unternehmensgröße* bietet beispielsweise folgende vier Auswahlmöglichkeiten an:

- Kleinunternehmen,
- kleine Unternehmen,
- ---

habt, da es mehrere Lieferanten geben kann, aber MyCBR erst ab Version 3.0 multiple Kardinalitäten unterstützt (siehe auch Fußnote 125).

- mittlere Unternehmen und
- Großunternehmen.

Diese Kategorisierung der Unternehmensgröße richtet sich nach den Angaben des Statistischen Bundesamts Deutschland.¹²⁷

Auch die Klasse Versandprozess beinhaltet einen Slot (*verkehrsträger*) mit dem Wertetyp Symbol. Die Auswahlmöglichkeiten sind die gängigen Verkehrsträger¹²⁸ Straße, Schiene, Wasserstraße und Luft. Die Verkehrsträger stellen ein wesentliches Charakterisierungsmerkmal eines Versandprozesses dar.

Der Slot *warenart* bildet die wichtigste Eigenschaft der Klasse Ware. In diesem Slot wird die Ware einer Warengruppe zugeordnet und dadurch klassifiziert. Die Vorlage für die Festlegung der im Slot auswählbaren Warengruppen geschieht auf der Grundlage des Warenverzeichnisses für Außenhandelsstatistik¹²⁹ vom Statistischen Bundesamt Deutschland. Dieses Warenverzeichnis ist eine Erweiterung des international anerkannten Harmonisierten Systems¹³⁰ der Weltzollorganisation. Die Warengruppen sind in verschiedene Kategorien auf mehreren Hierarchieebenen aufgeteilt. Insgesamt gibt es 21 mit römischen Ziffern gekennzeichnete Abschnitte, wie beispielsweise „Waren pflanzlichen Ursprungs“ (Abschnitt II) oder „Unedle Metalle und Waren daraus“ (Abschnitt XV). Auf diese Abschnitte verteilen sich insgesamt 97 Kapitel, die den Warentyp weiter spezifizieren. Je Kapitel gibt es noch unterschiedlich viele weitere Spezifizierungen eine Hierarchieebene tiefer. Die Ebene der Unterkapitel (eine Ebene unter den 97 Kapiteln) bildet im Slot *warenart* die tiefste Kategorisierungsebene. Insgesamt befinden sich auf dieser Ebene 1108 Unterkapitel, die eine für die Entwicklung eines ersten Prototyps des ontologiegestützten CBR-Systems, ausreichend detaillierte Spezifizierung des Warentyps ermöglicht.¹³¹

Für spezielle Waren, die nach internationalen Vorschriften¹³² als Gefahrgut eingestuft werden, bietet der Slot *gefahrgutklasse* eine Einordnungsmöglichkeit in eine der neun verschiedenen Gefahrgutklassen.¹³³ Tabelle 4 zeigt die neun Gefahrgutklassen.

127) Einstufungen: Ein Kleinstunternehmen hat bis zu neun Beschäftigte und bis zu 2 Mio. € Jahresumsatz; ein kleines Unternehmen hat mehr als 9 und bis zu 49 oder mehr als 2 Beschäftigte und bis zu 10 Mio. € Jahresumsatz; ein mittleres Unternehmen hat bis zu 249 Beschäftigte und bis zu 50 Mio. € Jahresumsatz. Hat ein Unternehmen mehr als 249 Beschäftigte oder über 50 Mio. € Jahresumsatz, wird es als Großunternehmen eingestuft. Vgl. DESTATIS (2011).

128) Vgl. BISCHOF ET AL. (2009), S. 18 f.

129) Vgl. DESTATIS (2011).

130) Beim Harmonisierten System handelt es sich um eine Nomenklatur zur Bezeichnung und Codierung von Waren. Diese Nomenklatur umfasst ca. 5000 Warengruppen, die durch einen sechsstelligen Code bezeichnet werden. Das Übereinkommen zur Verwendung dieser von der Weltzollorganisation entwickelten Nomenklatur wird weltweit von mehr als 200 Verwaltungen angewendet, um u.a. Zolltarife festzulegen. Weltweit werden ca. 98% der gehandelten Güter mithilfe des Harmonisierten Systems klassifiziert. Vgl. EK (2011), o. S. und WZO (2011), o. S.

131) Zusätzlich könnte die hier verwendete Einteilung der Ware in Warengruppen anhand des Warenverzeichnisses für Außenhandelsstatistik vom Deutschen Bundesamt für Statistik noch um zwei weitere Hierarchieebenen erweitert werden. Dies würde zu insgesamt mehr als 14.000 Einträgen im Slot *warenart* führen. Auf diese Erweiterung wird aus Zeitgründen verzichtet. Zusätzlich würde dieses Vorhaben die Ausführung von Protégé stark verlangsamen. Der Effekt der Verlangsamung tritt bereits bei der hier verwendeten Lösung durch lange Ladezeiten auf.

132) Vgl. UNECE (2011), o. S.

Klasse	Gefahrgut
1	explosive Stoffe
2	Gase
3	entzündbare flüssige Stoffe
4	entzündbare feste Stoffe
5	brandfördernde Stoffe
6	giftige und infektiöse Stoffe
7	radioaktive Stoffe
8	ätzende Stoffe
9	verschiedene gefährliche Stoffe

Tabelle 4: Gefahrgutklassen

Die Klasse *Ort* wird hauptsächlich durch zwei Slots mit Wertetyp Symbol charakterisiert. Der Slot *weltregion* bietet 192 Staaten¹³⁴ und übergeordnete Regionen¹³⁵ zur Auswahl, die später für die Ähnlichkeitsbestimmung als Taxonomie hierarchisiert werden. Der Staat Brasilien ist beispielsweise in folgende Hierarchie von Regionen eingeordnet:

Amerika >> Lateinamerika >> Süd-Amerika >> Brasilien

Für die Logistik existiert mit den Handelszonen¹³⁶ (Slot: *handelszone*) noch eine weitere zur Charakterisierung von Orten hilfreiche Kategorisierung. Die hier verwendeten Handelszonen werden in der folgenden Tabelle dargestellt.

133) Im Slot *gefahrenklasse* sind auch die Untergruppen der Klassen enthalten, die der Übersichtlichkeit halber in der Tabelle ausgeblendet sind.

134) Hierbei handelt es sich um die 192 Mitgliedsstaaten der Vereinten Nationen, vgl. LEMKE (2008), S. 190 und UN (2011), o. S.

135) Die Einteilung der Staaten nach Regionen erfolgte nach dem Vorbild der Einteilung der Vereinten Nationen, vgl. UN (2011), o. S.

136) Unter einer Handelszone wird hier eine Kooperation von verschiedenen Staaten verstanden, die mit einem Freihandelsabkommen die wirtschaftliche Zusammenarbeit erleichtern.

CAN	Andengemeinschaft
EurAsEC	Eurasische Wirtschaftsgemeinschaft
EU	Europäische Union
NAFTA	Nordamerikanisches Freihandelsabkommen
CARICOM	Karibische Gemeinschaft
CACM	Zentralamerikanischer Gemeinsamer Markt
Mercosur	gemeinsamer Markt Südamerikas
ECOWAS	Westafrikanische Wirtschaftsgemeinschaft
CEMAC	Zentralafrikanische Wirtschafts- und Währungsgemeinschaft
SACU	Südafrikanische Zollunion
EAC	Ostafrikanische Gemeinschaft
COMESA	Gemeinsamer Markt für das Östliche und Südliche Afrika
SAARC	Südasiatische Vereinigung für regionale Kooperation
GCC	Golf-Kooperationsrat
ASEAN	Verband Südostasiatischer Nationen
PIF	Pacific Islands Forum

Tabelle 5: Handelszonen

Die Klasse SupplyChain enthält insgesamt elf Slots mit dem Wertetyp Symbol. Alle diese Slots stellen Einflussfaktoren auf die Art der Gestaltung einer Supply Chain dar. Hinter den Slots¹³⁷

- *massenleistungsfähigkeit,*
- *pünktlichkeit,*
- *regelmäßigkeit,*
- *schnelligkeit,*
- *sicherheit,*
- *kosten und*
- *flexibilität*

stehen Prioritäten, die angeben, in welchem Ausmaß der jeweilige Einflussfaktor die Gestaltung der Supply Chain beeinflusst. Die möglichen Ausprägungen dieser sieben Slots sind in der folgenden Tabelle dargestellt.

unwichtig	weniger wichtig	neutral	eher wichtig	sehr wichtig
-----------	-----------------	---------	--------------	--------------

Tabelle 6: Ausprägungen von Einflussfaktoren (Prioritäten)

137) Die Einflussfaktoren wurden aus der Übersicht in BISCHOF et al. (2009), S. 18 f., übernommen und um den Faktor „Kosten“ erweitert.

Die beiden Slots *beziehungVerkäuferZuKäufer* und *beziehungVerkäuferZuLieferant* spiegeln eventuell bestehende Verbindungen zwischen Verkäufer und Lieferant bzw. Käufer wieder. Laut Aussage des Logistikexperten ändern sich die Entscheidungspräferenzen bei der Gestaltung von Supply Chains, wenn beispielsweise Verkäufer und Käufer nicht voneinander getrennte Unternehmen sind, sondern miteinander verbunden sind (z.B. Unternehmen mit gleichem Mutterkonzern). Es ist beispielsweise auch möglich, dass Verkäufer und Käufer Teil des gleichen Unternehmens sind. Ein denkbare Beispiel hierfür wäre der Versand von Maschinen vom Thyssen-Krupp-Standort Essen zum neuen Werk in Brasilien.

Ein weiterer wichtiger Entscheidungsfaktor bei der Gestaltung von Supply Chains wird durch den Slot *lieferbedingungen* repräsentiert. Hierbei handelt es sich um die in einem Kaufvertrag vereinbarten Bedingungen, durch die festgelegt wird, welche Versandkosten der Käufer oder der Verkäufer zu tragen hat und wer im Schadensfall (Beschädigung, Verlust, Diebstahl etc.) das finanzielle Risiko trägt. Um die Lieferbedingungen im Slot für die Klasse SupplyChain abzubilden, wird eine Auswahl der aktuellen Incoterms¹³⁸ verwendet. Tabelle 7 zeigt eine Übersicht über die verwendeten Incoterms.

EXW	ab Werk
FCA	Frei Frachtführer
FAS	frei längsseits Schiff
FOB	frei an Bord
CFR	Kosten und Fracht
CIF	Kosten, Versicherung und Fracht bis zum Bestimmungshafen
CPT	Fracht bezahlt bis
CIP	Fracht und Versicherung bezahlt
DAT	Geliefert Terminal
DAP	Geliefert benannter Ort
DDP	Geliefert Zoll bezahlt

Tabelle 7: Incoterms 2010

Auch die Art der Zahlungsabwicklung kann Entscheidungen im Supply Chain Management beeinflussen, beispielsweise wenn der Verkäufer einen Spediteur zur Abwicklung der vereinbarten Zahlungsweise beauftragt.¹³⁹ In diesem Fall würde die gewählte Zahlungsart ggf. die Auswahl eines des Spediteurs beeinflussen. Im Slot *zahlungsabwicklung* sind die folgenden Zahlungsarten auswählbar:¹⁴⁰

- Vorauszahlung,
- Anzahlung,

138) Es handelt sich hierbei um die 7. Revision der von der Internationalen Handelskammer am 01.01.2011 implementierten Incoterms 2010, vgl. ICC (2011), o. S.

139) Vgl. BISCHOF et al. (2009), S. 406.

140) Vgl. BISCHOF et al. (2009), S. 406 f.

- Zahlung gegen Nachnahme,
- Zahlung mittels Akkreditiv,
- Dokumente gegen Kasse,
- Dokumente gegen Akzept,
- Zahlung nach Eingang der Ware und Rechnung sowie
- Zahlung nach Ablauf eines Zahlungsziels.

„Einfache“ Wertetypen:

Weitere in der Ontologie verwendeten, Wertetypen für Slots sind String, Float und Integer. Der Slot *ortsname* beinhaltet beispielsweise eine einfache Zeichenkette. Der einzige Slot mit dem Wertetyp Integer heißt *zeitfrist* und beschreibt die ggf. im Kaufvertrag angegebene Zeitfrist in Tagen bis zur Erfüllung der Pflicht des Verkäufers (Versand der Ware an den Käufer).

Die Klasse *Ware* besitzt verschiedene Slots des Wertetyp Float. Die Abmessungen (*breite*, *länge*, *höhe*) werden in Meter angegeben, die Menge je nach Warenart entweder in Stück oder in Kubikmeter. Das Gesamtgewicht wird in Kilogramm und der Gesamtwert in Euro beziffert. Die jeweiligen Einheiten werden in Protégé bei der Konfiguration der Slots im Feld „Documentation“ hinterlegt.

4.2.3.3 Erstellung von Ähnlichkeitsmaßstäben

4.2.3.3.1 Lokale Ähnlichkeitsmaßstäbe

Mit der Erstellung einer Ontologie ist der erste große Schritt getan, einen Prototyp für ein ontologiegestütztes CBR-System zu entwickeln. Die Ontologie bildet das Fundament in diesem System und definiert die Struktur des Fallwissens. Die Fälle als Grundlegendes Element des Case-Based Reasonings sind in den Begriffen der Ontologie beschrieben. Die nächste wichtige und erfolgsentscheidende¹⁴¹ Aufgabe im Entwicklungsprozess besteht darin, die Ähnlichkeitsbestimmung von Fällen zu ermöglichen. Hierfür werden zunächst lokale Ähnlichkeitsmaßstäbe bestimmt, um die Ähnlichkeit zwischen einzelnen Attributen¹⁴² zu berechnen. Diese lokalen Ähnlichkeitsmaßstäbe werden später zu globalen Ähnlichkeitsmaßstäben aggregiert.

Es ist anzumerken, dass die im Folgenden erarbeiteten Ähnlichkeitsmaßstäbe zum Großteil auf subjektiven Einschätzungen des Verfassers in Abstimmung mit der Meinung des Logistikexperten beruhen. Für die Zwecke der Machbarkeitsprüfung eines Prototyps wird dies als vollkommen ausreichend angesehen. Für die Weiterentwicklung des ontologiegestützten CBR-Systems auf Grundlage des hier vorgestellten Prototyps sind Ähnlichkeitsmaßstäbe, Gewichtungen sowie Ausprägungen von Slots problemlos anpassbar und erweiterbar.

141) Die Qualität der Problemlösung mit Case-Based Reasoning ist im höchsten Maße abhängig von der Art der Fallrepräsentation und der Gestaltung von klaren Beschreibungen von Ähnlichkeiten zwischen Fällen, vgl. AVRAMENKO/KRASLAWSKI (2008), S. 71.

142) Unter „Attribut“ wird im Folgenden die Ausprägung eines Slots verstanden, die keine Relation darstellt. Relationen werden gesondert im Abschnitt 4.2.3.3.2 behandelt, da ihre Ähnlichkeit durch eine Aggregation mehrerer Attribute beeinflusst wird.

Die Ähnlichkeit sim_a zwischen den beiden Werten a^x und a^y eines Attributs a bei zwei miteinander vergleichenden Fällen x und y wird durch die Ähnlichkeitsfunktion

$$sim_a(a^x, a^y) \quad (4.1)$$

beschrieben, wobei a^x den Attributwert des Neu-Falls x und a^y den Attributwert des zum Vergleich herangezogenen Alt-Falls y repräsentiert. Die Ähnlichkeitsfunktion ist von Attribut zu Attribut unterschiedlich. Die verschiedenen Ausprägungen der Ähnlichkeitsfunktion werden im Folgenden erläutert.

Bei numerischen (quantitativen) Attributwerten gestaltet sich die Ähnlichkeitsbestimmung durch die Bestimmung des Abstands zwischen zwei Werten relativ einfach. Hierfür wird zunächst aus allen Ausprägungen eines Attributs in der Fallbasis die maximale Distanz $\max d$ errechnet und als Bezug für zwei zu vergleichende Attribute verwendet (siehe Formel 4.2)

$$sim_a(a^x, a^y) = 1 - \frac{|a^x - a^y|}{\max d}, \text{ mit } \max d = a^{\max} - a^{\min} \quad (4.2)$$

Ist im Beispiel *zeitfrist* die maximale Distanz aus der Fallbasis 45 Tage und sind $a^x = 7$ und $a^y = 14$, so beträgt die Ähnlichkeit zwischen den beiden Attributen:

$$sim_a(a^x, a^y) = 1 - \frac{|a^x - a^y|}{\max d} = 1 - \frac{|7 - 14|}{45} = 0,8\bar{4}$$

Auf diese Weise wird die Ähnlichkeit für die Ausprägung des Slots *zeitfrist* in der Klasse *SupplyChain* errechnet. Für die Float-Werte der Klasse *Ware* (*gesamtwicht*, *breite*, *länge*, *höhe*, *gesamtwert* und *menge*) ist diese Art der Berechnung aufgrund der vergleichsweise viel höheren maximalen Distanz weniger zu gebrauchen. Wird als Beispiel das *gesamtwicht* einer Warensendung betrachtet, so wäre die maximale Distanz gemessen in Kilogramm bei einem minimalen Wert von 5kg und einem maximalen Wert von 300.000 kg (z. B. Stahlcoils) bei 299.995 kg. Im Vergleich der zwei Attribute $a^x = 10$ und $a^y = 1.000$ beträgt die Ähnlichkeit:

$$sim_a(a^x, a^y) = 1 - \frac{|a^x - a^y|}{\max d} = 1 - \frac{|10 - 1.000|}{299.995} \approx 0,997$$

Trotz des hohen Unterschieds (a^y ist 100fach größer als a^x) wäre der Ähnlichkeitswert mit gerundeten 0,997 fast maximal. Dieses Phänomen beruht auf der Tatsache, dass die Abweichung der beiden Vergleichsattribute in Relation zur maximalen Distanz verschwindend gering ausfällt.

Die Lösung für dieses Problem ist die Errechnung der „relativen Distanz“. Hierbei wird die Ähnlichkeit zwischen zwei Attributen berechnet, indem der kleinere Wert direkt in Relation zum größeren Wert gesetzt wird. Ist der kleinere Wert beispielsweise halb so groß wie der größere Wert, so würde die Anwendung der folgenden Formel einen Ähnlichkeitswert von 0,5 ergeben:

$$sim_a(a^x, a^y) = \frac{\min(a^x, a^y)}{\max(a^x, a^y)} \quad (4.3)$$

Die Anwendung dieser Formel im Beispiel *gesamtwicht* (s.o.) ergäbe folgende Ähnlichkeit:

$$sim_a(a^x, a^y) = \frac{\min(a^x, a^y)}{\max(a^x, a^y)} = \frac{10}{1000} = 0,01$$

Im Vergleich von zwei Werten 10 kg und 1000 kg wird hier davon ausgegangen, dass die Ähnlichkeit gering ist. Der gleiche absolute Abstand (990 kg) würde bei zwei großen Attributwerten (z.B. $a^x = 20.000$ und $a^y = 20.990$) vergleichsweise geringer ins Gewicht fallen ($sim_a(a^x, a^y) \approx 0,95$) und somit nach Ansicht des Verfassers eine sinnvollere Aussage über die Ähnlichkeit zweier Attribute machen.

Da Protégé die relative Distanz nicht als Standardmaßstab anbietet, wird hier auf die Erweiterungsmöglichkeit der Skripterstellung zurückgegriffen. Die Umsetzung der relativen Distanz wird durch das Python-Skript aus Abbildung 14realisiert.

```
def sim(a_x, a_y):
    if a_x == a_y :
        return 1.0
    elif a_x == 0.0 or a_y == 0.0 :
        return 0.0
    elif a_x <= a_y :
        return a_x / a_y
    else :
        return a_y / a_x
```

Abbildung 14: Python-Skript für relative Distanz

Für die Attribute, die sich auf die Slots *name* und *ortsname* (Wertetyp String) beziehen, macht es wenig Sinn, einen speziellen Ähnlichkeitsmaßstab zu entwickeln.¹⁴³ Der Ähnlichkeitswert ist jeweils 1, wenn die Zeichenketten identisch sind, andernfalls ist der Ähnlichkeitswert 0. Für das Attribut Postleitzahl (Slot *postcode*) in der Klasse Ort wird ein Ähnlichkeitsmaßstab bestimmt, der auf der Annahme beruht, dass Postleitzahlen sich weltweit dadurch auszeichnen, dass die vorderen Stellen eine geografisch gröbere, überregionale Einteilung beschreiben und die Genauigkeit der geografischen Einordnung mit jeder Stelle zunimmt. Zwei zu vergleichende, gleichlange Postleitzahlen werden Stelle für Stelle miteinander verglichen, der Ähnlichkeitswert erhöht sich iterativ mit jeder weiteren Übereinstimmung, bis die erste Differenz auftaucht. Bei einer Differenz wird der Vergleich nicht mit den weiteren Stellen fortgeführt, da diese nicht mehr relevant sind.¹⁴⁴ Die iterative Erhöhung des Ähnlichkeitswerts geschieht degressiv und beginnt mit dem Wert 0,5 bei Übereinstimmung der ersten Stelle. Der Ähnlichkeitsmaßstab für die Postleitzahlen lässt sich, wie schon die relative Distanz (s.o.), nicht mit den Vorgaben von Protégé verwirklichen, sodass wieder ein Python-Skript aushelfen muss. Der Quellcode für dieses Skript ist in Abbildung 15 dargestellt.

143) Würde hier ein Ähnlichkeitsmaßstab entwickelt, der einen Ähnlichkeitswert größer als 0 wiedergibt für den Fall, dass die zu vergleichenden Zeichenketten gemeinsame Wortteile enthalten, so wäre beispielsweise die Ähnlichkeit zwischen den Namen „Deutsche Bank“ und „Deutsche Bahn“ sehr hoch, da sich nur die letzten beiden Buchstaben unterscheiden. Dieser Effekt wäre für den Zweck des Vergleichs zweier Akteure weniger wünschenswert wenn, wie im Beispiel, die Ähnlichkeit keine Aussagekraft hat.

144) Beispielsweise würde es bei einem Vergleich der deutschen Postleitzahlen 45127 (Region Essen) und 49127 (Region Osnabrück) keinen Sinn machen, die letzten zwei Stellen zu untersuchen, da durch den Unterschied in der zweiten Stelle bereits unterschiedliche Regionen gekennzeichnet sind und jede weitere Stelle nur die Orte in der jeweiligen Region detaillierter beschreibt.

```

def sim(a_x, a_y):
    if len(a_x) != len(a_y):
        return 0.0
    length = len(a_x)
    current = range(length)
    s = 0.5
    rtn = 0
    for i in range(0, length):
        if a_x[i] == a_y[i]:
            rtn = rtn + s
            s = ((1 - rtn) / 2)
        else:
            return rtn
    return 1.0

```

Abbildung 15: Python-Skript für Postleitzahlen

Symbolische Attributwerte bestehen zwar ebenfalls aus Begriffen (Strings), jedoch können die Ähnlichkeiten zwischen zwei Attributwerten trotz unterschiedlicher Begriffe bestehen, z.B. sind die Regionen „Barbados“ und „Jamaika“ durch die gemeinsame Lage in der Karibik ähnlich. Auf der anderen Seite sind Attribute nicht gleich ähnlich, nur weil sie Attributwerte haben. Um Attributwerte, die sich auf Slots mit dem Wertetyp Symbol beziehen, dennoch vergleichbar machen zu können, werden Tabellen und Taxonomien als Ähnlichkeitsmaßstäbe verwendet.

Wird ein Ähnlichkeitsmaßstab durch eine Tabelle verkörpert, wird für jede Kombinationsmöglichkeit zweier Attributwerte, der zugehörige Ähnlichkeitswert in einer Tabelle hinterlegt. Dabei steht der Zeilenindex für die Ausprägung des einen und der Spaltenindex für die Ausprägung des anderen zu vergleichenden Attributwerts. Die Tabelle ist symmetrisch, sodass Zeilen und Spalten vertauscht werden können:¹⁴⁵

$$sim_a(a^x, a^y) = tab_a(a^x, a^y) = tab_a(a^y, a^x) \quad (4.4)$$

Für die Attribute der sieben Slots der Klasse *SupplyChain*, die Prioritäten für die Gestaltung von Supply Chains repräsentieren (*massenleistungsfähigkeit*, *pünktlichkeit*, *regelmäßigkeit*, *schnelligkeit*, *sicherheit*, *kosten* und *flexibilität*) wird jeweils die Entscheidungstabelle aus Tabelle 8 als Ähnlichkeitsmaßstab verwendet.

145) Vgl. BEIBEL (2011), S. 160.

	unwichtig	weniger wichtig	neutral	eher wichtig	sehr wichtig
unwichtig	1	0,75	0,50	0,25	0
weniger wichtig	0,75	1	0,75	0,50	0,25
neutral	0,50	0,75	1	0,75	0,5
eher wichtig	0,25	0,50	0,75	1	0,75
sehr wichtig	0	0,25	0,5	0,75	1

Tabelle 8: Ähnlichkeitstabelle Prioritäten

Werden in einem Beispiel die Attributwerte des Slots *schnelligkeit* verglichen und gelten $a^x = \text{"eher wichtig"}$ sowie $a^y = \text{"sehr wichtig"}$ so lautet der Ähnlichkeitswert laut Tabelle 8: $tab_a(a^x, a^y) = 0,75$.

	getrenntes Unternehmen	selbes Unternehmen	verbundenes Unternehmen
getrenntes Unternehmen	1	0	0
selbes Unternehmen	0	1	0,5
verbundenes Unternehmen	0	0,5	1

Tabelle 9: Ähnlichkeitstabelle Unternehmensbeziehung

Ein weiterer tabellarischer Ähnlichkeitsmaßstab beschreibt die Ähnlichkeit für Attribute der Slots *beziehungVerkäuferZuKäufer* und *beziehungVerkäuferZuLieferant* und ist in Tabelle 9 dargestellt. Hierbei wird angenommen, dass zwischen der Beziehungsart „selbes Unternehmen“ und „verbundenes Unternehmen“ eine gewisse Ähnlichkeit (Wert 0,5) besteht.

Die Unternehmensgröße (Klasse *Unternehmen*, Slot *unternehmensgröße*) wird ebenfalls durch einen tabellarischen Ähnlichkeitsmaßstab bewertet. Auffällig ist hier, dass sich der Attributwert „Großunternehmen“ stark von den drei übrigen Attributwerten abgrenzt. Dies ist durch den großen „Abstand“ von den kleineren Unternehmen hinsichtlich Kennziffern wie Mitarbeiter und Jahresumsatz zu erklären (siehe auch Fußnote 127).

	Kleinst- unternehmen	kleines Unternehmen	mittleres Unternehmen	Groß- unternehmen
Kleinstunternehmen	1	0,9	0,5	0
kleines Unternehmen	0,9	1	0,6	0,1
mittleres Unternehmen	0,5	0,6	1	0,5
Großunternehmen	0	0,1	0,5	1

Tabelle 10: Ähnlichkeitstabelle Unternehmensgröße

Die Slots *nutztVerkehrsträger* und *handelszone* werden auch durch Tabellen in ihrer Ähnlichkeit bewertet. Jedoch ist bei der Handelszone nur die exakte Übereinstimmung zweier Attributwerte mit der Ähnlichkeit 1,0 bewertet, alle weiteren Vergleichspaare erhalten den Ähnlichkeitswert 0. Analog dazu sieht die Tabelle zur Ähnlichkeit der Verkehrsträger aus, mit der Ausnahme, dass zwischen den Attributwerten „Binnenschifffahrt“ und „Seeschifffahrt“ (wegen der offensichtlichen Gemeinsamkeiten der Beförderungsmittel und -wege) ein Ähnlichkeitswert von 0,5 eingesetzt wird.

Der lokale Ähnlichkeitsmaßstab für den Slot *zahlungsabwicklung* wird ebenfalls durch eine Tabelle erstellt. Hierbei werden die auf Vertrauen vom Käufer zum Verkäufer beruhenden Möglichkeiten „Vorauszahlung“ und „Anzahlung“ im Vergleich mit dem Ähnlichkeitswert 0,75 bewertet. Das Gleiche gilt für das entgegengesetzte Nichtvertrauen vom Verkäufer zum Käufer („Zahlung nach Eingang der Ware und Rechnung“ und „Zahlung nach Ablauf eines Zahlungsziels“). Die vier weiteren Möglichkeiten zur Zahlungsabwicklung bedürfen spezieller fachkundiger Mitwirkung eines Spediteurs¹⁴⁶ und werden im Vergleich zu anderen Zahlungsmöglichkeiten jeweils mit dem Ähnlichkeitswert 0 bewertet.

Bei symbolischen Attributwerten mit größeren Auswahlmöglichkeiten und einer kategorisierbaren Struktur bietet sich die Verwendung einer Taxonomie als Ähnlichkeitsmaßstab an. Eine Taxonomie ist ein hierarchisches Modell, um Begriffe eines bestimmten Bereichs nach bestimmten Kriterien zu klassifizieren, d.h. sie in bestimmte Kategorien einzuordnen¹⁴⁷. Die bekannten Ähnlichkeiten werden in der Baumstruktur einer Taxonomie an den Knoten hinterlegt, die untergeordnete Knoten besitzen. Der Ähnlichkeitswert bei einem Vergleich zweier Attribute a^x und a^y wird in der Taxonomie ermittelt, indem der Ähnlichkeitswert am nächsten gemeinsamen übergeordneten Knoten abgelesen wird. Der Ähnlichkeitsmaßstab lautet:

$$sim_a(a^x, a^y) = tax_a(a^x, a^y) \quad (4.5)$$

Je tiefer sich dieser gemeinsame Knoten in der Hierarchie der Taxonomie befindet, desto höher ist der Ähnlichkeitswert zweier zu vergleichender Attributwerte.

Der Slot *lieferbedingungen* enthält Attribute, für die eine Darstellung durch eine Taxonomie sinnvoll möglich ist (siehe Abbildung 16). Die 11 Lieferbedingungen (Incoterms 2010) lassen sich un-

146) Vgl. BISCHOF et al. (2009), S. 407 ff.

147) Vgl. „Taxonomie“ in KOSCHNIK (1993).

terteilen in allgemeingültige und nur die Schifffahrt betreffende Lieferbedingungen. Diese Unterscheidung bildet die erste Abzweigung in der baumartigen Struktur der Taxonomie.

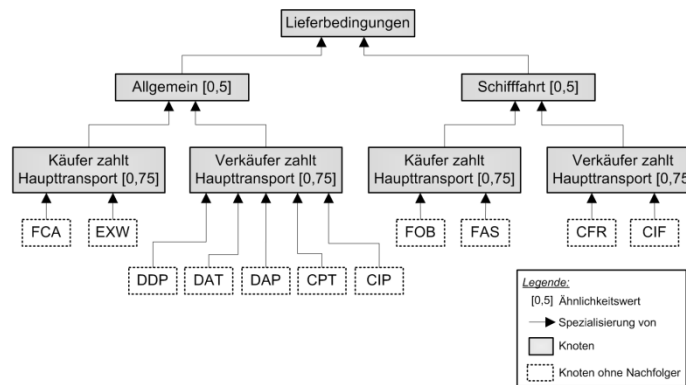


Abbildung 16: Taxonomie für Lieferbedingungen

Die Zahl in den eckigen Klammern steht für den Ähnlichkeitswert, der am jeweiligen Knoten hinterlegt ist. Die Lieferbedingungen *FOB* und *CIF* haben beispielsweise den Knoten *Schifffahrt* als nächsten gemeinsamen übergeordneten Knoten, was somit im Vergleich den Ähnlichkeitswert 0,5 ergibt. Eine Hierarchieebene tiefer wird nach dem Träger der Hauptkosten (Verkäufer bzw. Käufer) unterschieden. Die beiden Lieferbedingungen *CFR* und *CIF* erhalten im Vergleich einen Ähnlichkeitswert von 0,75.

Auch der Ähnlichkeitsmaßstab für Attribute des Slots *weltregion* wird durch eine Taxonomie realisiert. Die 192 Mitgliedsstaaten der UN (Vereinte Nationen) sind hierbei die Knoten ohne Nachfolger. Die Regionen (z.B. Mittelamerika) sind die kategorisierenden Knoten und haben Ähnlichkeitswerte zugeordnet. In Abbildung 17 wird ein kleiner Ausschnitt aus der Taxonomie abgebildet.

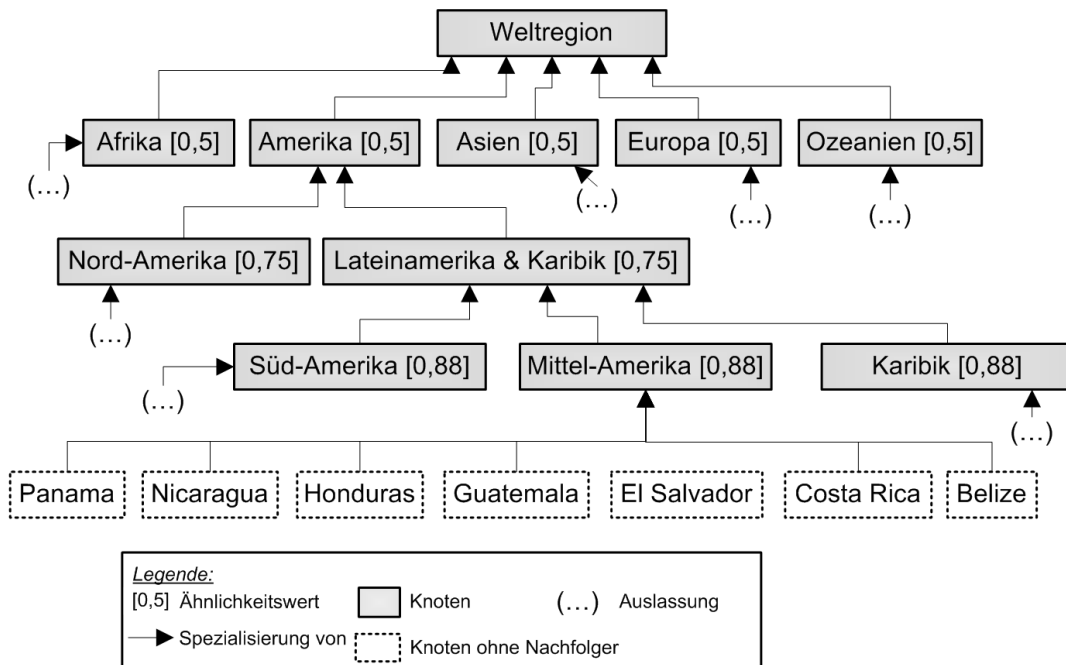


Abbildung 17: Auszug aus der Taxonomie für Weltregionen

Besonders umfangreich gestaltet sich die Taxonomie für die Ähnlichkeitsbestimmung der *warenart* der Klasse Ware. In Abbildung 18 ist ein sehr kleiner Ausschnitt dieser Taxonomie, wie sie in Protégé dargestellt wird, abgebildet.

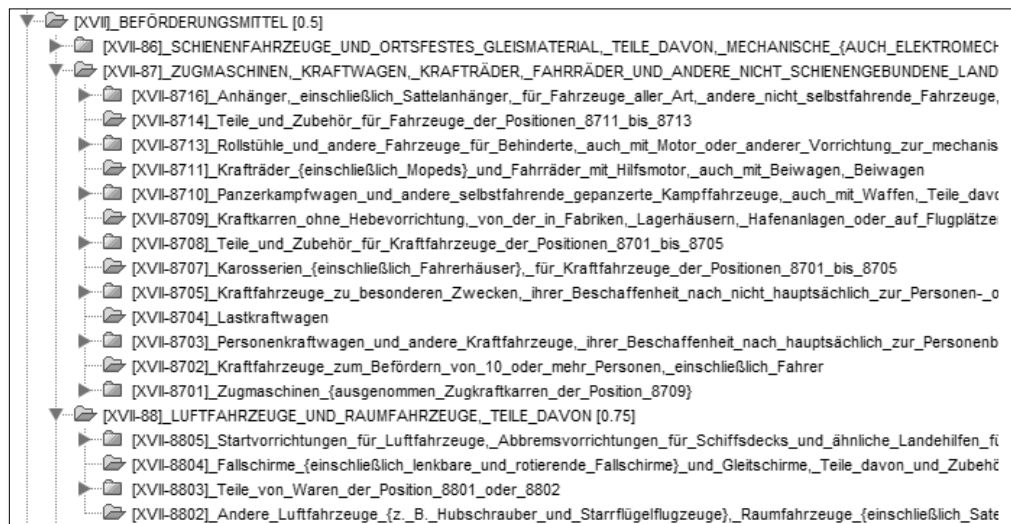


Abbildung 18: Protégé-Screenshot: Auszug aus Warenart-Taxonomie

Die Ähnlichkeitswerte wurden bei dieser Taxonomie einheitlich vergeben. Auf der ersten Hierarchieebene, in der sich die 21 Abschnitte des Warenkatalogs nach Vorlage des Statistischen Bundesamts befinden, erhält jeder Knoten den Ähnlichkeitswert 0,5. Auf der zweiten Ebene befinden sich die 97 Kapitel, die im Harmonisierten System durch die ersten beiden Stellen gekennzeichnet sind (Kapitel 1 beginnt z.B. mit „01“). Der Ähnlichkeitswert der Kapitel beträgt 0,75. Auf der dritten Ebene befinden sich die Unterkapitel, die durch die dritte bis vierte Stelle im Code des Harmonisierten Systems gekennzeichnet sind. Die 1108 Unterkapitel sind in der hier erstellten Taxonomie die Knoten ohne Nachfolger, d.h., zwei Attribute des Slots *warentyp*, die auf dasselbe Unterkapitel verweisen, besitzen die Ähnlichkeit 1. Diese Lösung ist noch nicht zufriedenstellend, da die drei weiteren Hierarchieebenen fehlen (siehe Fußnote 131). Um dieses Problem zu umgehen, wird ein Skript erstellt (im Slot *taricNr*), das anhand des zehnstelligen TARIC-Codes¹⁴⁸ die Stellen vergleicht, die für die drei fehlenden Hierarchieebenen stehen. Das Skript ist in Abbildung 19 dargestellt.

```
def sim(q, c):
    if len(q) != 10 or len(c) != 10:
        return 0.0
    if q[0]+q[1]+q[2]+q[3] != c[0]+c[1]+c[2]+c[3]:
        return 0.0
    if int(q[4])*10+int(q[5]) != int(c[4])*10+int(c[5]):
        return 0.0
    if int(c[6])*10+int(c[7]) != int(q[6])*10+int(q[7]):
        return 0.5
    if int(c[8])*10+int(c[9]) != int(q[8])*10+int(q[9]):
        return 0.75
    else:
        return 1.0
```

Abbildung 19: Skript für Ähnlichkeitsbestimmung von TARIC-Nummern

Dieses Skript errechnet nur dann einen Wert größer 0 für zwei zu vergleichende Attribute, wenn die Warenart in der Taxonomie identisch ist.

148) TARIC ist die Bezeichnung des in der Europäischen Union gemeinsam verwendeten Zolltarifs. Die Codierung baut auf dem Harmonisierten System auf (Codes mit sechs Stellen) und erweitert dieses um vier weitere Stellen. Vgl. ZOLL (2011), o. S.

Analog zu den bisher vorgestellten Taxonomien wird für den Slot *gefahr gutklasse* ebenfalls eine Taxonomie angelegt, die sich auf maximal zwei Hierarchieebenen beschränkt: die neun Gefahrgutklassen der UN-Wirtschaftskommission für Europa (UNECE) und ihre jeweiligen Unterklassen.¹⁴⁹

Der Ähnlichkeitsmaßstab für Relationen (Slots mit dem Wertetyp Instance) ist durch die Ähnlichkeit der verglichenen Instanzen gegeben. Die Vorgehensweise zur Ähnlichkeitsmessung von Instanzen wird im folgenden Abschnitt erläutert.

4.2.3.3.2 Globale Ähnlichkeitsmaßstäbe

Im vorigen Abschnitt wurden die lokalen Ähnlichkeitsmaßstäbe für jeweils einzelne Attribute von Slots betrachtet. Diese gilt es für den Vergleich von Instanzen einer Klasse nun zu globalen Ähnlichkeitsmaßstäben zu aggregieren.

Der Ähnlichkeitsmaßstab für Relationen wurde im vorangegangenen Abschnitt bewusst noch nicht behandelt, da es sich hierbei nicht um ein einzelnes zu vergleichendes Attribut handelt, sondern um eine Aggregation der Attribute der relational verknüpften Instanzen sowie eines allgemeinen Ähnlichkeitsmaßstabs der zugehörigen Klassen handelt. Die Ähnlichkeitsbestimmung zwischen zwei relational verknüpften Instanzen z_x und z_y lässt sich durch folgende Formel beschreiben:

$$sim_r(r^{z_x}, r^{z_y}) = \frac{\sum_{i \in I} (w_i \cdot sim_a(a_i^{z_x}, a_i^{z_y}))}{\sum_{i \in I} w_i} \cdot sim_k(k^{z_x}, k^{z_y}) \quad (4.6)$$

Die Ähnlichkeit der zwei relational verknüpften Instanzen wird mithilfe der Summe der mit den Gewichten w_i der gewichteten Ähnlichkeitswerte $sim_a(a_i^{z_x}, a_i^{z_y})$ für jeweils zwei Attributswerte $a_i^{z_x}$ und $a_i^{z_y}$ des Attributs i , die zu den beiden verglichenen Instanzen z_x bzw. z_y gehören, und zwar für alle Attribute $i \in I$ mit I als der Indexmenge für alle Attribute, über welche die beiden Instanzen z_x und z_y gemeinsam verfügen.¹⁵⁰ Hierbei steht I für die Gesamtheit aller gemeinsamen Attribute der relational verknüpften Instanzen und w_i für die jeweilige Attributgewichtung. Da auch Instanzen unterschiedlicher Klassen verglichen werden können, wird mit $sim_k(k^{z_x}, k^{z_y})$ ein Ähnlichkeitsmaßstab für den Vergleich von Klassen eingeführt. Dieser bezieht sich auf die Klassenhierarchie der Ontologie und wird durch eine Taxonomie bestimmt. Werden beispielsweise Instanzen der Klassen Person und Unternehmen miteinander verglichen, beträgt der Ähnlichkeitswert $sim_k(k^{z_x}, k^{z_y}) = 0,5$.

149) Siehe UNECE (2011), o. S.

150) Die gewichtete Summe bietet die gebräuchlichste Methode zur Aggregation von Attributwerten. Sie ist einfach bestimmbar und skalierbar. Der Gefahr der willkürlichen Belegung der Gewichte ist hier mit einer Absprache der Werte mit dem Logistikexperten begegnet worden. Für den Fall, dass die späteren Testergebnisse des CBR-Systems Diskrepanzen aufkommen lassen, können die Gewichte problemlos angepasst werden. Vgl. auch BEIBEL (2011), S. 184.

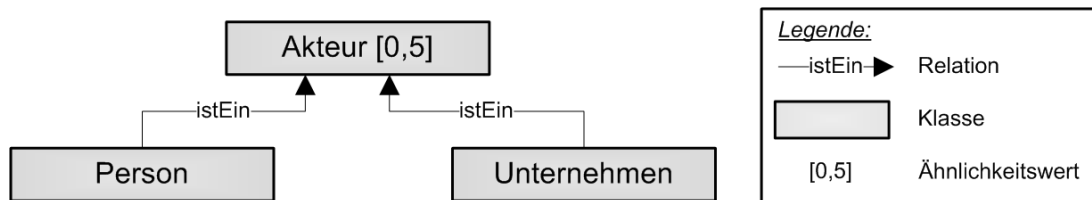


Abbildung 20: Ähnlichkeitswert für die Subklassen zur Klasse Akteur

Im nächsten Schritt können nun die lokalen Ähnlichkeitsmaßstäbe der einzelnen Attribute sim_a sowie die Ähnlichkeitsmaßstäbe der Relationen sim_r zu einem globalen Ähnlichkeitsmaßstab sim_g aggregiert werden, der zwei Fälle x und y (Instanzen der Klasse *SupplyChain*) miteinander vergleicht. Die Indexmengen I und J stehen für die Gesamtheit aller Attribute bzw. Relationen der zu vergleichenden Fälle. Hieraus ergibt sich die folgende Formel für den globalen Ähnlichkeitsmaßstab:¹⁵¹

$$sim_g(x, y) = \frac{\sum_{i \in I} (w_i \cdot sim_a(a_i^x, a_i^y)) + \sum_{j \in J} (w_j \cdot sim_r(r_j^{z_x}, r_j^{z_y}))}{\sum_{i \in I} w_i + \sum_{j \in J} w_j} \quad (4.7)$$

Mithilfe dieser Formel werden im CBR-System neu eingegebene Fälle mit Alt-Fällen verglichen. Um den globalen Ähnlichkeitsmaßstab zu vervollständigen, fehlen noch die Gewichtungen der Attribute und Relationen, die in Protégé durch Slots repräsentiert werden. Die Gewichtungen sind zunächst subjektiv ausgewählt und anschließend mit dem Logistikexperten abgestimmt worden. Die folgenden beiden Tabellen zeigen die für Attribute und Relationen gewählten Gewichtungen:

151) Die hier vorgestellte Formel gilt für die Ähnlichkeitsmessung von zwei Instanzen aus derselben Klasse *SupplyChain*, die im hier entwickelten CBR-System einen Fall repräsentiert. Es ließen sich auch problemlos Instanzen anderer Klassen der Ontologie vergleichen, allerdings müsste dann die globale Ähnlichkeit sim_g zusätzlich noch mit der Klassenähnlichkeit sim_k multipliziert werden, falls die zu vergleichenden Instanzen unterschiedliche Klassen instanzieren.

Klasse	Slot	Typ	Gewichtung
SupplyChain	beziehungVerkäuferZuKäufer	Attribut	1
	beziehungVerkäuferZuLieferant	Attribut	1
	flexibilität	Attribut	1
	kosten	Attribut	1
	lieferbedingungen	Attribut	2
	massenleistungsfähigkeit	Attribut	1
	pünktlichkeit	Attribut	1
	regelmäßigkeit	Attribut	1
	schnelligkeit	Attribut	1
	sicherheit	Attribut	1
	zeitfrist	Attribut	2
	befördertWare	Relation	8
	beginntIn	Relation	6
	endetIn	Relation	6
	hatKäufer	Relation	1
	hatVerkäufer	Relation	1
zahlungsabwicklung	Attribut	1	
Akteur	name	Attribut	1
	istLieferantFür	Relation	0
	ort	Attribut	1
Unternehmen	unternehmensgröße	Attribut	1

Tabelle 11: Gewichtungen der Attribute und Relationen (Teil 1)

Klasse	Slot	Typ	Gewicht
Ware	menge	Attribut	1
	breite	Attribut	1
	höhe	Attribut	1
	länge	Attribut	1
	gesamtgewicht	Attribut	2
	gesamtwert	Attribut	2
	taricNr	Attribut	1
	warenart	Attribut	2
	warenbezeichnung	Attribut	0
Ort	handelszone	Attribut	2
	ortsname	Attribut	1
	postcode	Attribut	1
	weltregion	Attribut	8

Tabelle 12: Gewichtungen des Retrieval -Algorithmus (Teil 2)

4.2.3.4 Erstellung des Retrieval-Algorithmus

Mithilfe der Formel für den globalen Ähnlichkeitsmaßstab kann nun das Retrieval des ähnlichsten Alt-Falls aus der Fallbasis begonnen werden. Der Algorithmus für das Retrieval wird im Folgenden durch sequentielle Darstellung der Arbeitsschritte dargestellt.

- 1) Der Algorithmus beginnt mit der Eingabe eines Neu-Falls in das CBR-System. Ein Neu-Fall wird eingegeben, indem zunächst eine Instanz der Klasse *SupplyChain* angelegt wird und anschließend die Ausprägungen aller benötigten¹⁵² Slots definiert werden.
- 2) Es wird eine Ergebnisliste angelegt, in die die Ergebnisse der Vergleiche des Neu-Falls mit den Fällen aus der Fallbasis gespeichert werden.
- 3) Für alle Fälle aus der Fallbasis wird anhand der Formel 4.7 die Ähnlichkeit zum Neu-Fall berechnet. Das Ergebnis (Ähnlichkeitswert) wird in die Ergebnisliste einsortiert (die Ergebnisliste ist absteigend nach Ähnlichkeitswerten sortiert).
- 4) Nach Abschluss der Vergleiche wird das oberste Element aus der Ergebnisliste als ähnlichster Fall aus der Fallbasis ausgegeben.
- 5) Das Retrieval ist beendet.

152) Unter „benötigte Slots“ werden hier die Slots verstanden, die eine Kardinalität von mindestens 1 aufweisen. Slots, die im Neu-Fall undefiniert bleiben, werden bei der Ähnlichkeitsberechnung im Vergleich mit den Alt-Fällen ignoriert. Wenn im umgekehrten Fall ein Slot im Neu-Fall definiert wird, der im Vergleichsfall undefiniert ist, wird die lokale Ähnlichkeit auf 0 gesetzt.

4.2.4 Test

4.2.4.1 Anlegen einer Fallbasis mit Testfällen

Die Grundlage für die Funktionalität des Case-Based Reasonings, ist das Vorhandensein einer Fall-datenbank mit Alt-Fällen aus der Anwendungsdomäne. Für den Test des Prototyps wurden mithilfe des Logistikexperten fünf Fälle aus der Praxis akquiriert, die auf der Basis der Ontologie beschrieben und in Protégé eingegeben wurden. Mit dieser geringen Anzahl von Fällen wird nicht beabsichtigt, die Grundgesamtheit des Domänenwissens des Supply Chain Managements repräsentativ abzubilden, es soll vielmehr eine qualitative Retrieval-Technik zu spezifischen Fällen getestet werden. Im Folgenden werden die Testfälle hinsichtlich ihrer wesentlichen Merkmale umschrieben, die genauen Ausprägungen werden im Anhang tabellarisch aufgeführt.

Der erste Fall beschreibt die Lieferung von speziellen Verbrennungsöfen von ThyssenKrupp Deutschland mit Versandort Essen zum Produktionsstandort in Brasilien. Die Verbrennungsöfen werden zunächst per LKWs zum Duisburger Hafen transportiert. Von dort geht es per Binnenschiff nach Bremerhaven und anschließend über den Pazifik nach Rio de Janeiro. Verkäufer und Käufer sind in diesem Fall Teil desselben Unternehmens.

Der zweite Fall behandelt den Versand von Mischtrommeln für Betonmischfahrzeuge von Belgrad (Serbien) zu einem KFZ-Hersteller nach Charleston (USA). Der Verkäufer hat den Unternehmenssitz in Koblenz. Dieser Fall eignet sich besonders gut für das Case-Based Reasoning, da das hierin enthaltene Problemlösungswissen nicht mathematisch-logisch darstellbar ist, sondern auf speziellen Erfahrungen mit dem Umgang mit Behörden und Logistikdienstleistern in Serbien beinhaltet. Die nach Angaben des Logistikexperten „vernünftigste“ Versandroute wäre die Verschiffung am nächstgelegenen Mittelmeerhafen mit Ziel US-amerikanische Ostküste (Norfolk). Die Erfahrung in diesem Fall hat aber gezeigt, dass es schneller und kostengünstiger ist, die Güter zunächst mit LKWs nach Hamburg zu transportieren und von dort aus den Schiffsweg zu nutzen. Die Gründe hierfür sind unter anderem die langsam arbeitenden Behörden in den Hafenstädten der Region (in Kroatien und Montenegro) sowie die geringe Frequentierung der Häfen.

Im dritten Fall werden große Stahlträger für eine Koksofenbatterie von Linz (Österreich) nach Shanghai transportiert. Nach dem Transport über die Straße nach Antwerpen wird die Ware per Schiff weiter nach China transportiert. Verkäufer und Käufer sind getrennte Unternehmen.

Der vierte Fall beschreibt den Transport von Stahlcoils von Duisburg (durch ThyssenKrupp) nach Madrid. Der Transport wird ohne Verkehrsträgerwechsel per Eisenbahn abgewickelt. Verkäufer und Käufer sind getrennte Unternehmen. Es handelt sich hierbei um einen regelmäßig stattfindenden Transport, bei dem hoher Wert auf geringe Kosten und Pünktlichkeit gelegt wird.

Der fünfte Fall behandelt einen Transport von MP3-Playern vom Produzenten in Malaysia bis zum deutschen Käufer. Der Versandweg setzt sich zusammen aus dem Schiffsweg von Kuala Lumpur bis nach Dubai, einem Flug weiter nach Frankfurt und zuletzt dem Transport auf der Straße bis zum Käufer nach Köln.

4.2.4.2 Test des Retrievals für das Case-based Reasoning

Anhand der im vorigen Abschnitt beschriebenen Testfälle wird der Retrieval-Mechanismus des CBR-Prototyps auf seine Funktionalität getestet. Hierbei gilt es nachzuprüfen, ob die Berechnungen

der globalen Ähnlichkeitswerte anhand der im Abschnitt 4.2.3.3 erstellten Ähnlichkeitsmaßstäbe vom System korrekt getätigt werden.

Im Testverfahren werden die fünf Testfälle untereinander verglichen. In jedem der insgesamt fünf Testdurchgänge wird jeweils einer der fünf Testfälle als Kopie einen Abfragefall darstellen und mit den Fällen der Falldatenbank verglichen und somit auch mit sich selbst. Als Ergebnis ergibt sich die unten abgebildete Kreuztabelle, aus der die Ähnlichkeitswerte aller 25 Paarvergleiche abgelesen werden können.

	Fall1	Fall2	Fall3	Fall4	Fall5
Fall1	1	0,46	0,43	0,44	0,24
Fall2	0,46	1	0,39	0,35	0,25
Fall3	0,43	0,39	1	0,41	0,21
Fall4	0,44	0,35	0,41	1	0,36
Fall5	0,24	0,25	0,21	0,36	1

Tabelle 13: Retrievalergebnisse

Jeder Retrievalvorgang wurde zur Überprüfung der Zuverlässigkeit des CBR-Systems zehnmal wiederholt, hierbei wurden Abfragezeiten zwischen 0,062 und 0,093 Sekunden¹⁵³ erreicht. Die Ergebnisse waren bei allen zehn Wiederholungen identisch, womit das Kriterium der Zuverlässigkeit erfüllt ist.

Um die Korrektheit der Ähnlichkeitsberechnung zu überprüfen, wird willkürlich ein Paarvergleich ausgewählt, der manuell berechnet wird. Das Ergebnis muss mit dem Ergebnis des CBR-Retrievals (Tabelle 13) übereinstimmen.

Die manuelle Ähnlichkeitsberechnung wird anhand des Vergleichs von Fall 1 mit Fall 2 durchgeführt. In Tabelle 14 sind die beiden Fälle gegenübergestellt.¹⁵⁴

153) Getestet wurde auf einem 32-Bit System mit einem 2-GHz-Prozessor und 2 GB RAM.

154) Die Attributwerte wurden für die Ansicht in Tabellenform ggf. abgekürzt, um die Übersichtlichkeit zu bewahren.

SupplyChain	Fall1	Fall2	Ähnlichkeitswert
Slot	Wert	Wert	
Attribute			
beziehungVerkäufer ZuKäufer	<i>verbundenes Unternehmen</i>	<i>getrenntes Unternehmen</i>	0,0000
beziehungVerkäufer ZuLieferant	--	--	--
flexibilität	<i>unwichtig</i>	<i>eher wichtig</i>	0,2500
kosten	<i>eher wichtig</i>	<i>neutral</i>	0,7500
lieferbedingungen	<i>FAS</i>	<i>FOB</i>	0,7500
massenleistungsfähigkeit	<i>sehr wichtig</i>	<i>neutral</i>	0,5000
pünktlichkeit	<i>eher wichtig</i>	<i>sehr wichtig</i>	0,7500
regelmäßigkeit	<i>unwichtig</i>	<i>unwichtig</i>	1,0000
schnelligkeit	<i>eher wichtig</i>	<i>unwichtig</i>	0,2500
sicherheit	<i>neutral</i>	<i>unwichtig</i>	0,5000
zeitfrist	23	18	0,8649
zahlungsabwicklung	<i>nach Warenein- gang (Rechnung)</i>	<i>Zahlungsziel</i>	0,7500
Relationen ¹⁵⁵			
befördertWare	<i>Verbrennungsöfen</i>	<i>Mischtrommeln für Betonfahrzeuge</i>	0,3398
beginntIn	<i>Essen</i>	<i>Belgrad</i>	0,4000
endetIn	<i>Rio De Janeiro</i>	<i>Charleston</i>	0,3333
hatKäufer	<i>ThyssenKrupp CSA</i>	<i>Charleston Motors</i>	0,4444
hatVerkäufer	<i>ThyssenKrupp AG</i>	<i>Huckmann GmbH</i>	0,4444

Tabelle 14: Testergebnis

Die berechneten lokalen Ähnlichkeiten des Fallvergleichs können nun mithilfe der Formel (4.7) zu einer globalen Ähnlichkeit zwischen Fall 1 und Fall 2 aggregiert werden:

$$sim_g(Fall1, Fall2) = \frac{\sum_{i=1}^{11} (w_i \cdot sim_a(a_i^{Fall1}, a_i^{Fall2})) + \sum_{j=1}^5 (w_j \cdot sim_r(r_j^{ZFall1}, r_j^{ZFall2}))}{\sum_{i=1}^{11} w_i + \sum_{j=1}^5 w_j}$$

155) Die genaue Berechnung der Ähnlichkeitswerte für die Relationen wird im Anhang aufgeführt.

$$\sum_{i=1}^{11} (w_i \cdot \text{sim}_a(a_i^{\text{Fall1}}, a_i^{\text{Fall2}}))$$

$$\approx 1 \cdot 0 + 1 \cdot 0,25 + 1 \cdot 0,75 + 2 \cdot 0,75 + 1 \cdot 0,5 + 1 \cdot 0,75 + 1 \cdot 1 + 1 \cdot 0,25 + 1 \cdot 0,5 + 2 \cdot 0,8649 + 1 \cdot 0,75 \approx \mathbf{7,9798}$$

$$\sum_{j=1}^5 (w_j \cdot \text{sim}_r(r_j^{\text{Fall1}}, r_j^{\text{Fall2}})) \approx 8 \cdot 0,3398 + 6 \cdot 0,4 + 6 \cdot 0,3 + 1 \cdot 0,4 + 1 \cdot 0,4 \approx \mathbf{8,0072}$$

$$\rightarrow \text{sim}_g(\text{Fall1}, \text{Fall2}) \approx \frac{7,9798 + 8,0072}{13 + 22} \approx \mathbf{0,4568}$$

Die manuelle Errechnung der Ähnlichkeit von Fall 1 und Fall 2 ergibt somit 0,4568 (in Tabelle 13 auf 0,46 gerundet) und stimmt mit dem Retrieval-Ergebnis des CBR-Prototyps überein.

5 Fazit und Ausblick

Die im Supply Chain Management auftretenden Effizienzverluste durch das Fehlen einer gemeinsamen Sicht der jeweiligen Akteure auf die Gesamtheit der Wertschöpfungsprozesse haben den Bedarf an einer innovativen Lösung zur Verbesserung der Organisation und Gestaltung von Supply Chains geweckt. Im Zuge dessen hat das Verbundprojekt OrGoLo zum Ziel, eine interaktive „Web-2.0“ Kollaborationsplattform zu entwickeln. Bestandteil dieser Plattform ist unter anderem ein wissensbasiertes und lernfähiges Assistenztool auf Grundlage von Case-Based Reasoning.

In der vorliegenden Arbeit wurde ein Prototyp entwickelt, der die Machbarkeit eines Assistenztools zur Unterstützung der Gestaltung von Supply Chains aufzeigt. Die Ontologie zur Beschreibung der Anwendungsdomäne erhebt hierbei keineswegs einen Anspruch auf „Vollständigkeit“ (wenn diese überhaupt erreichbar ist), sondern greift einen Ausschnitt aus der Welt der Logistik heraus, der auf sehr abstrakte Art und Weise eine Supply Chain mit seinen Akteuren, diversen Einflussfaktoren, der versendeten Ware und beliebig vielen Teilabschnitten von Versandprozessen beschreibt. Diese Abstrahierung ist notwendig gewesen, um für den im Rahmen einer Diplomarbeit angemessenen Detaillierungsgrad für die Entwicklung eines ontologiegestützten CBR-Systems zu erhalten. Die Darstellung des Domänenwissens in Form einer Ontologie macht es problemlos möglich, aufbauend auf den hier vorgestellten Ergebnissen ein umfassendes CBR-System zu entwickeln oder die Ergebnisse an anderer Stelle wiederzuverwenden.

In dieser Arbeit wurde Entwicklungsprozess eines Prototyps für ontologiegestütztes Case-based Reasoning in Supply Chains beschrieben. Die Entwicklung basierte auf der Integration der zwei Entwicklungsumgebungen Protégé und MyCBR, die jeweils als Werkzeug für die Gestaltung der ontologischen Wissensrepräsentation bzw. deren Nutzung für das Case-Based Reasoning verwendet wurden. Die Komplexität von Supply Chains zeigten vor allem der Software MyCBR ihre Grenzen auf. So war es nicht möglich Klassenbeziehungen mit einer variablen Kardinalität (größer 1) abzubilden. Ebenso kam es zu Problemen bei Zyklen in den Relationen der Ontologie¹⁵⁶. Des Weiteren

156) Ein Zyklus findet statt, wenn eine Klasseninstanz über mehrere Relationen auf sich selbst verweist. Das wäre z. B. der Fall, wenn die Versandprozesse die inversen Relationen *istNachfolgerVon* und *istVorgängerVon* erhalten hätten.

kam es zu etlichen Systemabstürzen, die jedesmal zur Folge hatten, dass sämtliche erstellten (und auch zuvor gespeicherten) Ähnlichkeitsmaßstäbe gelöscht wurden. Aus diesen Gründen rät der Verfasser von der Benutzung einer Version 2.x¹⁵⁷ von MyCBR zur Entwicklung eines CBR-Systems für Supply Chains ab. Vielmehr bietet sich eine individuelle Lösung an, um den komplexen Ansprüchen gerecht werden zu können.

Der Test mit einer exemplarischen Fallbasis von fünf Testfällen zeigte valide Ergebnisse hinsichtlich der zu lokalen und globalen Ähnlichkeitsmaßstäben erstellten Formeln.

Die in dieser Arbeit entwickelte Ontologie mit dem darauf aufbauenden Case-based-Reasoning-System zeigt die Machbarkeit qualitativen Retrievals im Supply Chain Management auf. Die strukturierte Form der Ontologie bietet die Möglichkeit, den hier behandelten und auf wesentliche Aspekte reduzierten Teilausschnitt der Anwendungsdomäne nach Belieben zu erweitern oder anzupassen. So wurden beispielsweise in dieser Arbeit keine parallelen Zuliefer-Versandprozesse berücksichtigt, die dem eigentlichen Versand des Endprodukts vorausgehen können. Des Weiteren sind die Klassen in ihren Eigenschaften noch weiter differenzierbar.

Weiterhin ist die Sicht auf die Supply Chain nicht auf die hier verwendete Vorgabe der Klasse *SupplyChain* beschränkt. Die globalen Ähnlichkeiten erlauben beispielsweise auch das Retrieval von Instanzen der anderen in der Ontologie definierten Klassen. Für die Entwicklung einer Benutzeroberfläche für die Nutzung der Case-based-Reasoning Funktionalität durch diverse Logistik-Akteure wäre eine individuell anpassbare Suchmöglichkeit umsetzbar, auch die Attributgewichtungen sind anpassbar und können auf die Prioritäten der Suchanfrage abgestimmt werden.

157) Auf eine Anfrage an das Entwicklerteam von MyCBR wurde als Alternative die Verwendung der neueren Version 3.0 angeraten, die jedoch (noch?) nicht mit Protégé kompatibel ist.

6 Literaturverzeichnis

Vorbemerkungen:

- Alle Quellen werden im Literaturverzeichnis wie folgt aufgeführt: In der ersten Zeile wird der *Referenztitel* der Quelle angegeben. Er entspricht der Form, die im Text Verwendung findet, wenn auf die Quelle hingewiesen wird.
- Bei der Vergabe der Referenztitel wird bei *einem* Autor dessen Nachname, gefolgt von dem Erscheinungsjahr der Quelle in Klammern, verwendet. Existieren *zwei* oder *drei* Autoren, werden diese getrennt von einem Schrägstrich („/“) aufgeführt. Bei mindestens *vier* Autoren werden nur die ersten drei Autoren mit dem Zusatz „et al.“ aufgeführt.
- Zu *Internetquellen* wird die dafür verantwortliche Instanz aufgeführt. Dies können sowohl natürliche als auch juristische Personen sein. Zu den Internetquellen werden die zum Zugriffsdatum gültige Internetadresse (URL) und das Zugriffsdatum angegeben.

AAMODT/PLAZA (1994)

Aamodt, A.; Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. In: AI Communications. IOS Press, Vol. 7 (1994), No. 1, S. 39-59.

AHA/BRESLOW (1997)

Aha, D. W.; Breslow, L. A.: Refining Conversational Case Libraries. In: Leake, D. B.; Plaza, E. (eds.): Case-Based Reasoning Research and Development, Proceedings of the second International Conference on Case-Based Reasoning ICCBR-97. Berlin et al. 1997, S. 267-279.

ALLEN (2008)

Allen, R. E.: Human ecology economics: a new framework for global sustainability. New York (USA) 2008.

AVRAMENKO/KRASLAWSKI (2008)

Avramenko, Y.; Kraslawski, A.: Case Based Design – Applications in Process Engineering. Berlin et al. 2008.

BALZERT (2009)

Balzert, H.: Lehrbuch der Softwaretechnik – Basiskonzepte und Requirements Engineering. 3. Aufl., Heidelberg 2009.

BARBIER/PEARCE (2000)

Barbier, E.; Pearce, D. W.: Blueprint for a sustainable economy. London (UK) 2000.

BECKMANN (2007)

Beckmann, M.: Corporate Social Responsibility und Corporate Citizenship. Halle 2007.

BEIBEL (2011)

Beißel, S.: Ontologiestütztes Case-Based Reasoning – Entwicklung und Beurteilung semantischer Ähnlichkeitsindikatoren für die Wiederverwendung natürlichsprachig repräsentierten Projektwissens. Dissertation an der Universität Duisburg-Essen. Wiesbaden 2011.

BERGMANN/SCHAAF (2003)

Bergmann, R.; Schaaf, M.: Structural Case-Based Reasoning and Ontology-Based Knowledge Management: A Perfect Match? In: Journal of Universal Computer Science, Vol. 9 (2003), Nr. 7, S. 608-626.

BERGMANN et al. (1999)

Bergmann, R.; Althoff, K.-D.; Breen, S.; Göker, M.; Manago, M.; Traphöner, R.; Wess, S.: Developing Industrial Case-Based Reasoning Applications – The INCREA Methodology. Berlin et al. 1999.

BISCHOF ET AL. (2009)

Bischof, K. D.; Meister, H.; Pyell, G.; Roj, G.; Stadler, U.; Wagner, G.: Leistungserstellung in Spedition und Logistik. 11. Aufl., Troisdorf 2009.

BITRAN/GURUMURTHIO/SAM (2006)

Bitran, G. R.; Gurusurthio, S.; Sam, S. L.: Emerging Trends in Supply Chain Governance. Working Report, MIT Sloan School of Management. Cambridge 2006.

BLANZIERI/EBRANATI (2000)

Blanzieri, E.; Ebranati, A.: Supporting Tourism Culture via CBR. In: Blanzieri, E.; Portinale, L. (eds.): Advances in Case-Based Reasoning: 5th European Workshop, EWCBR 2000. Berlin et al. 2000, S. 359-370.

BMIR (2011)

Standford Center for Biomedical Informatics Research. Online-Präsenz für das Ontologie-Tool Protégé. URL: „<http://protege.stanford.edu/>“, letzter Zugriff am 01.06.2011.

BUTCHER/LALWANI/MANGAN (2008)

Butcher, T.; Lalwani, C.; Mangan, J.: Global logistics and supply chain management. Chichester 2008.

CHEETHAM/WATSON (2006)

Cheetham, W.; Watson, I.: Fielded applications of case-based reasoning. In: The Knowledge Engineering Review, Vol. 20 (2006), Issue 3, S. 321-323.

CONWAY (1997)

Conway, M. A.: Cognitive Models of Memory. King's Lynn (UK) 1997

CURBACH (2009)

Curbach, J.: Die Corporate-social-responsibility-Bewegung. Wiesbaden 2009.

DALY (2007)

Daly, H. E.: Ecological economics and sustainable development: selected essays of Herman Daly. 2. Aufl., Cheltenham 2007.

DESTATIS (2011)

Deutsches Bundesamt für Statistik. Online-Präsenz. URL: „<http://www.destatis.de/>“, letzter Zugriff am 09.06.2011.

DFKI (2011)

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH. Online-Präsenz für das CBR-Tool MyCBR. URL: „<http://www.myCBR-project.net/>“, letzter Zugriff am 27.05.2011.

DITTMANN/SCHÜTTE/ZELEWSKI (2003)

Dittmann, L., Schütte, R., Zelewski, S.: Darstellende Untersuchung philosophischer Probleme mit Ontologien. In: Freyberg, K.; Petsche, H.-J.; Klein, B. (Hrsg.): Knowledge Management and Philosophy. Proceedings of the WM 2003 Workshop on Knowledge Management and Philosophy, 02.-04.04.2003 in Luzern, CEUR Workshop Proceedings, Vol. 85 (2003), 3. Beitrag. Kaiserslautern 2003.

EK (2011)

Europäische Kommission. Online-Präsenz. URL: „http://ec.europa.eu/taxation_customs/customs/customs_duties/tariff_aspects/harmonised_system/index_de.htm“, letzter Zugriff am 09.06.2011.

ERLER (2006)

Erler, M.: Platon. München 2006.

FENSEL (2004)

Fensel, D.: Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce. 2. Aufl., Berlin et al. 2004.

FERGER (2009)

Ferger, F.: Nachhaltigkeit in Lieferketten – eine ökonomische Analyse. Norderstedt 2009.

FREUDENTHALER (2008)

Freudenthaler, B.: Case-based Reasoning (CBR). Saarbrücken 2008.

FINNIE/SUN (2003)

Finnie, G.; Sun, Z.: R^5 model for case-based reasoning. In: Knowledge-Based Systems, Vol. 16 (2003), No. 1, S. 59-65.

GAŠEVIĆ/DJURIĆ/DEVEDŽIĆ (2006)

Gašević, D.; Djurić, D.; Devedžić, V.: Model Driven Architecture and Ontology Development. Berlin et al. 2006.

GOMEZ-PEREZ/FERNANDEZ-LOPEZ/CORCHO-GARCIA (2004)

Gomez-Perez, A.; Fernandez-Lopez, M.; Corcho-Garcia, O.: Ontological Engineering: With examples from the areas of Knowledge Management, e-Commerce and the Semantic Web, 2. Aufl., London 2004.

GÖKER/ROTH-BERGHOFER (1999)

Göker, M.; Roth-Berghofer, T.: Development and Utilization of a Case-Based Help-Desk Support System in a Corporate Environment. In: Althoff, K.-H.; Bergmann, R.; Branting, L. K. (Eds.): Case-Based Reasoning Research and Development, Proceedings of the 3rd International Conference on Case-Based Reasoning ICCBR-99. Berlin et al. 1999. S. 132-146.

GÖRZ/ROLLINGER/SCHNEEBERGER (2003)

Görz, G.; Rollinger, C.-R.; Schneeberger, J.: Handbuch der Künstlichen Intelligenz. 4. Aufl., München et al. 2003.

GRUBER(1993)

Gruber, T. R.: A Translation Approach to Portable Ontology Specifications. In: Knowledge Acquisition Vol. 5 (1993), No. 2, S. 199-220.

GRUBER (1995)

Gruber, T. R.: Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In: International Journal Human-Computer Studies, Vol. 43 (1995), S. 907-928.

GUARINO (1998)

Guarino, N.: Formal Ontology and Informations Systems. In: Guarino, N. (ed.): Formal Ontology in Information Systems. Proceedings of FOIS '98. Amsterdam 1998, S. 3-15.

GUARINO/OBERLE/STAAB (2009)

Guarino, N.; Oberle, D., Staab, S.: What Is an Ontology? In: Staab, S.; Studer, R. (eds.): Handbook on Ontologies. 2. Aufl. Dordrecht et al. 2009, S. 1-17.

HEIDER (1996)

Heider, R.: Troubleshooting CFM 56-3 Engines for the Boeing 737 using CBR and Data-Mining. In: Faltings, B.; Smith, I. (Eds.): Advances in case based reasoning: Third European Workshop, EWCBR-96. Lausanne 1996. S. 512-518.

HINKLE/TOOMEY (1995)

Hinkle, D.; Toomey, C.: Applying Case-Based Reasoning to Manufacturing. In: AI Magazine Vol. 16 (1995), No 1. 1995, S. 65-73.

HORRIDGE (2011)

Horridge, M.: A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools. Edition 1.3. Online-Präsenz. URL: „<http://www.co-ode.org/resources/tutorials/ProtegeOWLTutorial.pdf>“, letzter Zugriff am 31.05.2011.

ICC (2011)

Internationale Handelskammer. Online-Präsenz. URL: „<http://www.icc-deutschland.de/index.php?id=46>“, letzter Zugriff am 10.06.2011

JANSEN (2008a)

Jansen, L.: Ein neues Bild von Ontologie. In: Jansen, L.; Smith, B. (eds.): Biomedizinische Ontologie: Wissen strukturieren für den Informatikeinsatz. Zürich 2008, S. 229-232.

JANSEN (2008b)

Jansen, L.: Kategorien: Die top level Ontologie. In: Jansen, L.; Smith, B. (Eds.): Biomedizinische Ontologie: Wissen strukturieren für den Informatikeinsatz. Zürich 2008. S. 85-112.

KATER (2011)

Kater, D.: Case-Based Reasoning in Supply Chains: Qualitatives Case-Retrieval. Diplomarbeit am Institut für Produktion und Industrielles Informationsmanagement der Universität Duisburg-Essen. Unveröffentlichte Qualifizierungsarbeit: Essen 2011.

KLÜVER/SCHMIDT/STOICA (2006)

Klüver, J.; Schmidt, J.; Stoica, C.: Mathematisch-logische Grundlagen der Informatik. Bochum 2006.

KOLODNER (1992)

Kolodner, J. L.: An Introduction to Case-Based Reasoning. In: Artificial Intelligence Review Vol. 6 (1992), No. 1, S. 3-34.

KOLODNER (1997)

Kolodner, J.: Case-Based Reasoning. 5. Aufl., San Mateo (USA) 1997.

KOSCHNIK (1993)

Koschnik, W. J.: Standardwörterbuch für die Sozialwissenschaften (Band 2), Paris 1993.

KUCKARTZ (2010)

Kuckartz, U.: Einführung in die computergestützte Analyse qualitativer Daten. 3. Aufl. Wiesbaden 2010.

LEAKE (1996)

Leake, D. B.: Case-Based Reasoning: Experiences, Lessons and Future Directions. Menlo Park 1996.

LEMKE (2008)

Lemke, C.: Internationale Beziehungen: Grundkonzepte, Theorien und Problemfelder. 2. Aufl., München 2008.

LIGGESMEYER (2009)

Liggemeyer, P.: Software-Qualität – Testen, Analysieren und Verifizieren von Software. 2. Aufl., Heidelberg 2009.

MAASS (2009)

Maass, W.: Elektronische Wissensmärkte: Handel von Information und Wissen über digitale Netze. Wiesbaden 2009.

MALLIN (2009)

Mallin, C. A.: Corporate social responsibility: a case study approach. Cheltenham 2009.

NILSSON/SOLLENBORN (2004)

Nilsson, M.; Sollenborn, M.: Advancements and Trends in Medical Case-Based Reasoning: An Overview of Systems and System Development. In: Barr, V.; Markov, Z. (eds.): Proceedings of the 17th International Florida Artificial Intelligence Research Society Conference. Miami Beach 2004, S. 178-183.

NOY/MCGUINNESS (2001)

Noy, N. F.; McGuinness, D. L.: Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 und Stanford Medical Informatics Technical Report SMI-2001-0880. Stanford 2001.

NUSCHELER (2009)

Nuscheler, F.: Good governance: ein universelles Leitbild von Staatlichkeit und Entwicklung? INEF (Institut für Entwicklung und Frieden) Report Band 96. Duisburg 2009.

OKAMOTO/SATOH (1994)

Okamoto, S.; Satoh, K.: An Average Predictive Accuracy for the Nearest Neighbour Classifier. In: Haton, J.-P., Keane, Mark; Manago, Michel (eds.): Advances in Case-Based Reasoning, Second European Workshop, EWCBR-94. Chantilly 1994, S. 101-112.

ORGOLO (2011)

Produktblatt zum Projekt OrGoLo (Organisatorische Innovationen mit Good Governance in Logistik-Netzwerken) als Teil des „EffizienzCluster LogistikRuhr“. URL: „<http://www.uni-due.de/zlv/effizienzcluster/innovationen-mit-good-governance.php>“, letzter Zugriff am 17.07.2011.

ORO/RUFFOLO (2008)

Oro, E.; Ruffolo, M.: Towards a System for Ontology-Based Information Extraktion from PDF Documents. In: Meersman, R.; Tari, Z. (eds.): On the Move to Meaningful Internet Systems: OTM 2008. Berlin et al. 2008, S. 1482-1499.

PETERS (2010)

Peters, N.: Design of voluntary sustainability initiatives for supply chains. In: Bogaschewsky, R. et al. (Hrsg.): Supply Management Research: Aktuelle Forschungsergebnisse 2010. Wiesbaden 2010, S. 61-94.

PFUHL (2003)

Pfuhl, M.: Case-Based Reasoning auf der Grundlage relationaler Datenbanken – Eine Anwendung zur strukturierten Suche in Wirtschaftsnachrichten. Dissertation an der Philipps-Universität Marburg. Wiesbaden 2003.

RIESBECK/SCHANK (1989)

Riesbeck, C. K.; Schank, R. C.: Inside Case-Based Reasoning. New Jersey 1989.

Pinto/Martins (2004)

Pinto, H. S.; Martins, J. P.: Ontologies: How can They be Built? In: Knowledge and Information Systems Vol. 6 (2004), No. 4, S. 441-464.

PURVIS/PU (1998)

Purvis, L.; Pu, P.: COMPOSER: A Case Based Reasoning System for Engineering Design. In: Robotica archive Vol. 16 (1998), Issue 3, S. 285-295.

ROOME (1998)

Roome, N. J.: Sustainability strategies for industry: the future of corporate practice. Washington 1998.

ROSS (1989)

Ross, B. H.: Reminders in learning and instruction. In: Vosniadou, S.; Ortony, A. (eds.): Similarity and analogical reasoning. Cambridge 1989, S. 438-469.

ROTH-BERGHOFFER (2003)

Roth-Berghofer, T. R.: Knowledge Maintenance of Case-Based Reasoning Systems – The SIAM Methodology. Berlin 2003.

RUTKOWSKI (2005)

Rutkowski, L.: Computational Intelligence – Methods and Techniques. Warschau 2005.

SCHANK (1999)

Schank, R. C.: Dynamic Memory Revisited. Cambridge 1999.

SEURING/MÜLLER (2008)

Seuring, S.; Müller, M.: Core Issues in Sustainable Supply Chain Management-a Delphi Study. In: Business Strategy and the Environment Vol. 17 (2008), Issue 8, S. 455-466.

SMYTH ET AL. (2000)

Smyth, M. M.; Collins, A. F.; Morris, P. E.; Levy, P.: Cognition in Action, 2nd edition. Hove (UK) 2000.

SOMMER (2007)

Sommer, P.: Umweltfokussiertes Supply Chain Management: Am Beispiel des Lebensmittelsektors. Dissertation, Wiesbaden 2007.

STEIN ET AL. (2007)

Stein, B.; Suermann, M.; Büning, H. K.; Kelbassa, H.-W.; Reckmann, A.; Tellmann, R.; Thureau, C.; Wiegard, H.-G.: Fallbasiertes Schließen – Case-Based Reasoning, Grundlagen und Anwendung für Konstruktions- und Entwurfsaufgaben. Paderborn 2007.

STUCKENSCHMIDT (2009)

Stuckenschmidt, H.: Ontologien – Konzepte, Technologien und Anwendungen. Dordrecht et al. 2009.

STÜTZER (2007)

Stützer, S.: „Good Governance“: Modebegriff oder neues Entwicklungskonzept für fragile Staaten. Norderstedt 2007.

UN (2011)

United Nations. Online-Präsenz. URL: „<http://www.un.org/>“, letzter Zugriff am 09.06.2011.

UNECE (2011)

United Nations Economic Commission for Europe. Online-Präsenz. URL: „http://live.unece.org/trans/danger/publi/unrec/rev14/14files_e.html“, letzter Zugriff am 11.06.2011.

WATSON (2001)

Watson, I.: Applying Case-Based Reasoning – Techniques for Enterprise Systems. 5. Aufl., San Francisco 2001.

WATSON (1999)

Watson, I.: Case-based reasoning is a methodology not a technology. In: Knowledge-Based Systems Vol. 12 (1999), Issue 5-5, S. 303-308.

WEBER/WILD (2005)

Weber, B.; Wild, W.: Conversational Case-Based Reasoning Support for Business Process Management. In: Mixed-Initiative Problem-Solving Assistant - Papers from the AAAI Fall Symposium. Menlo Park 2005. S. 134-139.

WILHELM (2009)

Wilhelm, K.: Return on Sustainability. Indianapolis 2009.

WITTGENSTEIN/SCHULTE (2001)

Wittgenstein, L.; Schulte, J. (Hrsg.): Philosophische Untersuchungen: Kritisch-genetische Edition. Berlin 2001.

WZO (2011)

Weltzollorganisation. Online-Präsenz. URL: „http://www.wcoomd.org/home_hsoverviewboxes_hsharmonizedsystem.htm“, letzter Zugriff am 09.06.2011.

ZELEWSKI (1999)

Zelewski, S.: Ontologien zur Strukturierung von Domänenwissen – Ein Annäherungsversuch aus betriebswirtschaftlicher Perspektive. Arbeitsbericht Nr.3. Universität Essen 1999.

ZELEWSKI (2005)

Zelewski, S.: Einführung in das Themenfeld „Ontologien“ aus informations- und betriebswirtschaftlicher Perspektive. In: Zelewski, S.; Alan, Y.; Alparslan, A.; Dittmann, L.; Weichelt, T. (Hrsg.): Ontologiebasierte Kompetenzmanagementsysteme – Grundlagen, Konzepte, Anwendungen. Berlin 2005. S. 115-228.

ZELEWSKI (2011)

Zelewski, S.: Überblick über das Verbundprojekt OrGoLo – Organisatorische Innovationen mit Good Governance in Logistik-Netzwerken. Institut für Produktion und Industrielles Informationsmanagement, Universität-Duisburg-Essen. Essen 2011.

ZÖLLER-GREER (2002)

Zöller-Greer, P.: Softwareengineering für Ingenieure und Informatiker – Planung, Entwurf und Implementierung. Braunschweig/ Wiesbaden 2002.

ZOLL (2011)

Deutscher Zoll. Online-Präsenz. URL: „http://www.zoll.de/faq/faq_ezt_taric/index.html“, letzter Zugriff am 13.06.2011.

7 Anhang 1: Falldatenbank

Instanzen der Klasse *SupplyChain*

Fall 1	
Slot	Wert
Attribute	
beziehungVerkäuferZuKäufer	verbundenes Unternehmen
beziehungVerkäuferZuLieferant	
flexibilität	unwichtig
kosten	eher wichtig
lieferbedingungen	FAS--frei laengsseits Schiff - Free Alongside Ship
massenleistungsfähigkeit	sehr wichtig
pünktlichkeit	eher wichtig
regelmäßigkeit	unwichtig
schnelligkeit	eher wichtig
sicherheit	neutral
zeitfrist	23
Relationen	
befördertWare	Verbrennungsöfen 8417100000
beginntIn	Essen 45127
endetIn	Rio De Janeiro 20510-000
hatKäufer	ThyssenKrupp CSA
hatVerkäufer	ThyssenKrupp AG

Fall 2	
Slot	Wert
Attribute	
beziehungVerkäuferZuKäufer	getrenntes Unternehmen
beziehungVerkäuferZuLieferant	
flexibilität	eher wichtig
kosten	neutral
lieferbedingungen	FOB--frei an Bord - Free On Board
massenleistungsfähigkeit	neutral
pünktlichkeit	sehr wichtig
regelmäßigkeit	unwichtig
schnelligkeit	unwichtig
sicherheit	unwichtig
zeitfrist	18
Relationen	
befördertWare	Mischtrommeln für Betonfahrzeuge 8708999700
beginntIn	Belgrad 11000
endetIn	Charleston 29401
hatKäufer	Charleston Motors
hatVerkäufer	Huckmann GmbH

Fall 3	
Slot	Wert
Attribute	
beziehungVerkäuferZuKäufer	getrenntes Unternehmen
beziehungVerkäuferZuLieferant	
flexibilität	neutral
kosten	eher wichtig
lieferbedingungen	CIF--Kosten, Versicherung und Fracht bis zum Bestimmungshafen - Cost Insurance Freight
massenleistungsfähigkeit	neutral
pünktlichkeit	eher wichtig
regelmäßigkeit	weniger wichtig
schnelligkeit	weniger wichtig
sicherheit	weniger wichtig
zeitfrist	45
Relationen	
befördertWare	Stahlelemente für Koksofenbatterie 7220908000
beginntIn	Linz 4020
endetIn	Shanghai 200000
hatKäufer	Xyojiing Ltd
hatVerkäufer	Vöest

Fall 4	
Slot	Wert
Attribute	
beziehungVerkäuferZuKäufer	getrenntes Unternehmen
beziehungVerkäuferZuLieferant	
flexibilität	weniger wichtig
kosten	sehr wichtig
lieferbedingungen	EXW--ab Werk - EX Works
massenleistungsfähigkeit	weniger wichtig
pünktlichkeit	sehr wichtig
regelmäßigkeit	eher wichtig
schnelligkeit	sehr wichtig
sicherheit	unwichtig
zeitfrist	8
Relationen	
befördertWare	Stahlcoils 7207208000
beginntIn	Duisburg 47051
endetIn	Madrid 28000
hatKäufer	Steel Espaniol
hatVerkäufer	ThyssenKrupp AG

Fall 5	
Slot	Wert
Attribute	
beziehungVerkäuferZuKäufer	getrenntes Unternehmen
beziehungVerkäuferZuLieferant	
flexibilität	eher wichtig
kosten	sehr wichtig
lieferbedingungen	DDP--Geliefert Zoll bezahlt - Delivered Duty Paid
massenleistungsfähigkeit	sehr wichtig
pünktlichkeit	eher wichtig
regelmäßigkeit	neutral
schnelligkeit	eher wichtig
sicherheit	weniger wichtig
zeitfrist	12
Relationen	
befördertWare	MP3-Player 8519890000
beginntIn	Kuala Lumpur 56100
endetIn	Köln 54321
hatKäufer	Schneider AG
hatVerkäufer	MatAhmad Ltd

Instanzen der Klasse *Ware*

Ware Verbrennungsöfen 8417100000	
Slot	Wert
Attribute	
menge	20
breite	4
höhe	6
länge	4
gesamtgewicht	100000
gesamtwert	600000
taricNr	8417100000
warenart	[XVI-8417] nicht elektrische Industrie- und Laboratoriumsöfen, einschließlich Verbrennungsöfen
warenbezeichnung	Verbrennungsöfen

Ware Mischtrommeln fuer Betonfahrzeuge 8708999700	
Slot	Wert
Attribute	
menge	5
breite	2
höhe	2
länge	7
gesamtgewicht	100000
gesamtwert	25000
taricNr	8708999700
warenart	[XVII-8708] Teile und Zubehör für Kraftfahrzeuge der Positionen 8701 bis 8705
warenbezeichnung	Mischtrommeln für Betonfahrzeuge

Ware Stahlelemente für Koksofenbatterie 7220908000	
Slot	Wert
Attribute	
menge	20
breite	2
höhe	10
länge	15
gesamtgewicht	200000
gesamtwert	200000
taricNr	7220908000
warenart	[XV-7220] flachgewalzte Erzeugnisse aus nicht rostendem Stahl mit einer Breite von weniger als 600 mm
warenbezeichnung	Stahlelemente für Koksofenbatterie

Ware MP3-Player 8519890000	
Slot	Wert
Attribute	
menge	1000
breite	0,07
höhe	0,05
länge	0,1
gesamtgewicht	300
gesamtwert	12000
taricNr	8519890000
warenart	[XVI-8519] Tonaufnahmegeräte, Tonwiedergabegeräte, Tonaufnahme- und -wiedergabegeräte
warenbezeichnung	MP3-Player

Ware Stahlcoils 7207208000	
Slot	Wert
Attribute	
menge	50
breite	5
höhe	5
länge	20
gesamtgewicht	1500000
gesamtwert	300000
taricNr	7207208000
warenart	[XV-7207] Halbzeug aus Eisen oder nicht legiertem Stahl
warenbezeichnung	Stahlcoils

Instanzen der Klasse *Ort*

Ort Madrid 28000	
Slot	Wert
Attribute	
ortsname	Madrid
postcode	28000
weltregion	Spanien
handelszone	EU Europäische Union

Ort Bonn 53113	
Slot	Wert
Attribute	
ortsname	Bonn
postcode	53113
weltregion	Deutschland
handelszone	EU Europäische Union

Ort Koblenz 56068	
Slot	Wert
Attribute	
ortsname	Koblenz
postcode	56068
weltregion	Deutschland
handelszone	EU Europäische Union

Ort New York 10010	
Slot	Wert
Attribute	
ortsname	New York
postcode	10010
weltregion	USA
handelszone	NAFTA Nordamerikanisches Freihandelsabkommen

Ort Antwerpen 2000	
Slot	Wert
Attribute	
ortsname	Antwerpen
postcode	2000
weltregion	Belgien
handelszone	EU Europäische Union

Instanzen der Klasse *Person*

Person Stephen Miller	
Slot	Wert
Attribute	
name	Stephen Miller
Relationen	
istLieferantFür	
ort	Charleston 29401

Person Pedro Geromel	
Slot	Wert
Attribute	
name	Pedro Geromel
Relationen	
istLieferantFür	
ort	Rio De Janeiro 20510-000

Person Xing Do	
Slot	Wert
Attribute	
name	Xing Do
Relationen	
istLieferantFür	
ort	Shanghai 200000

Person Bernd Kullmann	
Slot	Wert
Attribute	
name	Bernd Kullmann
Relationen	
istLieferantFür	
ort	Köln 54321

Instanzen der Klasse *Unternehmen*

Unternehmen ThyssenKrupp CSA	
Slot	Wert
Attribute	
name	ThyssenKrupp CSA
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Rio De Janeiro 20510-000

Unternehmen ThyssenKrupp AG	
Slot	Wert
Attribute	
name	ThyssenKrupp AG
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Essen 45127

Unternehmen Spedition Müller	
Slot	Wert
Attribute	
name	Spedition Müller
unternehmensgröße	kleines Unternehmen
Relationen	
istLieferantFür	
ort	Essen 45127

Unternehmen Reederei H. Krug	
Slot	Wert
Attribute	
name	Reederei H. Krug
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Hamburg 20095

Unternehmen American Railways	
Slot	Wert
Attribute	
name	American Railways
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	New York 10010

Unternehmen Charleston Motors	
Slot	Wert
Attribute	
name	Charleston Motors
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Charleston 29401

Unternehmen Xyojiing Ltd	
Slot	Wert
Attribute	
name	Xyojiing Ltd.
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Shanghai 200000

Unternehmen Vöest	
Slot	Wert
Attribute	
name	Vöest
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Linz 4020

Unternehmen Deutsche Bahn	
Slot	Wert
Attribute	
name	Deutsche Bahn
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Frankfurt 60311

Unternehmen Benelux Ferries	
Slot	Wert
Attribute	
name	Benelux Ferries
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Antwerpen 2000

Unternehmen Lufthansa	
Slot	Wert
Attribute	
name	Lufthansa
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Frankfurt 60311

Unternehmen DHL	
Slot	Wert
Attribute	
name	DHL
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Bonn 53113

Unternehmen Schneider AG	
Slot	Wert
Attribute	
name	Schneider AG
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Köln 54321

Unternehmen Steel Espaniol	
Slot	Wert
Attribute	
name	Steel Espaniol
unternehmensgröße	Großunternehmen
Relationen	
istLieferantFür	
ort	Madrid 28000

Unternehmen Spedition Misevic	
Slot	Wert
Attribute	
name	Spedition Misevic
unternehmensgröße	Kleinstunternehmen
Relationen	
istLieferantFür	
ort	Belgrad 11000

Unternehmen Spedition Prater	
Slot	Wert
Attribute	
name	Spedition Prater
unternehmensgröße	Kleinstunternehmen
Relationen	
istLieferantFür	
ort	Wien 1100

Unternehmen Reederei D Krüger	
Slot	Wert
Attribute	
name	Reederei D Krüger
unternehmensgröße	mittleres Unternehmen
Relationen	
istLieferantFür	
ort	Duisburg 47051

Unternehmen Reederei B Schmitz	
Slot	Wert
Attribute	
name	Reederei B Schmitz
unternehmensgröße	mittleres Unternehmen
Relationen	
istLieferantFür	
ort	Bremerhaven 27568

Unternehmen Huckmann GmbH	
Slot	Wert
Attribute	
name	Huckmann GmbH
unternehmensgröße	mittleres Unternehmen
Relationen	
istLieferantFür	
ort	Koblenz 56068

Unternehmen Sasovic DOO	
Slot	Wert
Attribute	
name	Sasovic DOO
unternehmensgröße	mittleres Unternehmen
Relationen	
istLieferantFür	
ort	Belgrad 11000

Unternehmen MatAhmad Ltd	
Slot	Wert
Attribute	
name	MatAhmad Ltd
unternehmensgröße	mittleres Unternehmen
Relationen	
istLieferantFür	
ort	Kuala Lumpur 56100

Unternehmen Reederei KL	
Slot	Wert
Attribute	
name	Reederei KL
unternehmensgröße	mittleres Unternehmen
Relationen	
istLieferantFür	
ort	Kuala Lumpur 56100

Instanzen der Klasse *Versandprozess*

Versandprozess 1	
Slot	Wert
Attribute	
nutztVerkehrsträger	Strassenverkehr
Relationen	
istTeilVon	Fall 1
beginntIn	Essen_45127
endetIn	Duisburg_47051
hatVersender	ThyssenKrupp_AG
hatEmpfänger	Reederei_D_Krüger
hatFrachtführer	Spedition_Müller
hatSpediteur	Spedition_Müller
hatZollagent	

Versandprozess 2	
Slot	Wert
Attribute	
nutztVerkehrsträger	Binnenschifffahrt
Relationen	
istTeilVon	Fall 1
beginntIn	Duisburg_47051
endetIn	Bremerhaven_27568
hatVersender	Reederei_D_Krüger
hatEmpfänger	Reederei_B_Schmitz
hatFrachtführer	Reederei_D_Krüger
hatSpediteur	
hatZollagent	

Versandprozess 3	
Slot	Wert
Attribute	
nutztVerkehrsträger	Seeschifffahrt
Relationen	
istTeilVon	Fall 1
beginntIn	Bremerhaven_27568
endetIn	Rio_De_Janeiro_20510-000
hatVersender	Reederei_B_Schmitz
hatEmpfänger	ThyssenKrupp_CSA
hatFrachtführer	Reederei_B_Schmitz
hatSpediteur	
hatZollagent	Pedro_Geromel

Versandprozess 4	
Slot	Wert
Attribute	
nutztVerkehrsträger	Strassenverkehr
Relationen	
istTeilVon	Fall 2
beginntIn	Belgrad_11000
endetIn	Hamburg_20095
hatVersender	Spedition_Misevic
hatEmpfänger	Reederei_H_Krug
hatFrachtführer	Spedition_Misevic
hatSpediteur	Spedition_Misevic
hatZollagent	

Versandprozess 5	
Slot	Wert
Attribute	
nutztVerkehrsträger	Seeschiffahrt
Relationen	
istTeilVon	Fall 2
beginntIn	Hamburg_20095
endetIn	Norfolk_23501
hatVersender	Reederei_H_Krug
hatEmpfänger	American_Railways
hatFrachtführer	Reederei_H_Krug
hatSpediteur	
hatZollagent	Stephen_Miller

Versandprozess 6	
Slot	Wert
Attribute	
nutztVerkehrsträger	Eisenbahnverkehr
Relationen	
istTeilVon	Fall 2
beginntIn	Norfolk_23501
endetIn	Charleston_29401
hatVersender	American_Railways
hatEmpfänger	Charleston_Motors
hatFrachtführer	American_Railways
hatSpediteur	
hatZollagent	

Versandprozess 7	
Slot	Wert
Attribute	
nutztVerkehrsträger	Strassenverkehr
Relationen	
istTeilVon	Fall 3
beginntIn	Linz_4020
endetIn	Antwerpen_2000
hatVersender	Vöest
hatEmpfänger	Benelux_Ferries
hatFrachtführer	Spedition_Prater
hatSpediteur	Spedition_Prater
hatZollagent	

Versandprozess 8	
Slot	Wert
Attribute	
nutztVerkehrsträger	Seeschifffahrt
Relationen	
istTeilVon	Fall 3
beginntIn	Antwerpen_2000
endetIn	Shanghai_200000
hatVersender	Benelux_Ferries
hatEmpfänger	Xyojiing_Ltd
hatFrachtführer	Benelux_Ferries
hatSpediteur	
hatZollagent	Xing Do

Versandprozess 9	
Slot	Wert
Attribute	
nutztVerkehrsträger	Eisenbahnverkehr
Relationen	
istTeilVon	Fall 4
beginntIn	Duisburg_47051
endetIn	Madrid_28000
hatVersender	ThyssenKrupp_AG
hatEmpfänger	Steel_Espaniol
hatFrachtführer	Deutsche_Bahn
hatSpediteur	
hatZollagent	

Versandprozess 10	
Slot	Wert
Attribute	
nutztVerkehrsträger	Seeschiffahrt
Relationen	
istTeilVon	Fall 5
beginntIn	Kuala_Lumpur_56100
endetIn	Dubai_999041
hatVersender	MatAhmad_Ltd
hatEmpfänger	Lufthansa
hatFrachtführer	Reederei_KL
hatSpediteur	
hatZollagent	

Versandprozess 11	
Slot	Wert
Attribute	
nutztVerkehrsträger	Luftverkehr
Relationen	
istTeilVon	Fall 5
beginntIn	Dubai_999041
endetIn	Frankfurt_60311
hatVersender	Lufthansa
hatEmpfänger	DHL
hatFrachtführer	Lufthansa
hatSpediteur	
hatZollagent	Bernd_Kullmann

Versandprozess 12	
Slot	Wert
Attribute	
nutztVerkehrsträger	Strassenverkehr
Relationen	
istTeilVon	Fall 5
beginntIn	Frankfurt_60311
endetIn	Köln_54321
hatVersender	DHL
hatEmpfänger	Schneider_AG
hatFrachtführer	DHL
hatSpediteur	
hatZollagent	

8 Anhang 2: Testvergleich zwischen Fall 1 und Fall 2

SupplyChain	Fall1	Fall2	sim	w
Slot	Wert	Wert		
Attribute				
beziehungVerkäufer ZuKäufer	verbundenes Unternehmen	getrenntes Unternehmen	0,0000	1
beziehungVerkäufer ZuLieferant				1
flexibilität	unwichtig	eher wichtig	0,2500	1
kosten	eher wichtig	neutral	0,7500	1
lieferbedingungen	FAS--frei längsseits Schiff - Free Alongside Ship	FOB--frei an Bord - Free On Board	0,7500	2
massenleistungsfähigkeit	sehr wichtig	neutral	0,5000	1
pünktlichkeit	eher wichtig	sehr wichtig	0,7500	1
regelmäßigkeit	unwichtig	unwichtig	1,0000	1
schnelligkeit	eher wichtig	unwichtig	0,2500	1
sicherheit	neutral	unwichtig	0,5000	1
zeitfrist	23	18	0,8649	2
zahlungsabwicklung	Zahlung nach Eingang der Ware und Rechnung	Zahlung nach Ablauf eines Zahlungsziels	0,7500	1
Relationen				
befördertWare	Verbrennungsöfen 8417100000	Mischtrommeln fuer Betonfahrzeuge 8708999700	0,3398	8
beginntIn	Essen 45127	Belgrad 11000	0,4000	6
endetIn	Rio De Janeiro 20510-000	Charleston 29401	0,3333	6
hatKäufer	ThyssenKrupp CSA	Charleston Motors	0,4444	1
hatVerkäufer	ThyssenKrupp AG	Huckmann GmbH	0,4444	1

0,4568

Ware	Verbrennungsöfen 8417100000	Mischtrommeln für Betonfahrzeuge 8708999700	Ählichkeitswer	w
Slot	Wert	Wert		
Attribute				
menge	20	5	0,2500	1
breite	4	2	0,5000	1
höhe	6	2	0,3333	1
länge	4	7	0,5714	1
gesamtgewicht	100000	100000	1,0000	2
gesamtwert	600000	25000	0,0417	2
taricNr	8417100000	8708999700	0,0000	1
warenart	[XVI-8417] nicht elektrische Industrie- und Laboratoriumsöfen einschließ- lich Verbrennungsöfen	[XVII-8708] Teile und Zu- behör für Kraftfahrzeuge der Positionen 8701 bis 8705	0,0000	2
warenbezeichnung	Verbrennungsöfen	Mischtrommeln fuer Beton- fahrzeuge	0,0000	0

0,3398

Ort	Essen 45127	Belgrad 11000	sim	w
Slot	Wert	Wert		
Attribute				
ortsname	Essen	Belgrad	0,0000	1
postcode	45127	11000	0,0000	1
weltregion	Deutschland	Serbien	0,5000	8
handelszone	EU Europäische Union		0,0000	2

0,4000

Ort	Rio De Janeiro 20510-000	Charleston 29401	sim	w
Slot	Wert	Wert		
Attribute				
ortsname	Rio De Janeiro	Charleston	0,0000	1
postcode	20510-000	29401	0,0000	1
weltregion	Brasilien	USA	0,5000	8
handelszone	Mercosur Gemeinsamer Markt Süd- amerikas	NAFTA Nordamerika-nisches Frei- handelsabkommen	0,0000	2

0,3333

Unternehmen	ThyssenKrupp CSA	Charleston Motors	sim	w
Slot	Wert	Wert		
Attribute				
name	ThyssenKrupp CSA	Charleston Motors	0,0000	1
unternehmensgröße	Großunternehmen	Großunternehmen	1,0000	1
Relationen				
istLieferantFür				0
ort	Rio De Janeiro 20510-000	Charleston 29401	0,3333	1

0,4444

Unternehmen	ThyssenKrupp AG	Huckmann GmbH	sim	w
Slot	Wert	Wert		
Attribute				
name	ThyssenKrupp AG	Huckmann GmbH	0,0000	1
unternehmensgröße	Großunternehmen	mittleres Unternehmen	0,5000	1
Relationen				
istLieferantFür				0
ort	Essen 45127	Koblenz 56068	0,8333	1

0,4444

Ort	Essen 45127	Koblenz 56068	sim	w
Slot	Wert	Wert		
Attribute				
ortsname	Essen	Koblenz	0,0000	1
postcode	45127	56068	0,0000	1
weltregion	Deutschland	Deutschland	1,0000	8
handelszone	EU Europäische Union	EU Europäische Union	1,0000	2

0,8333

Autoren:

Dipl.-Inf. Martin Kowalski

E-Mail: martin.kowalski@pim.uni-due.de

Dipl.-Wirt.-Inf. Dennis Kater

E-Mail: rongricker@gmail.com

Impressum:

Institut für Produktion und
 Industrielles Informationsmanagement
 Universität Duisburg-Essen, Campus Essen
 Fakultät für Wirtschaftswissenschaften
 Universitätsstraße 9, 45141 Essen
 Website (Institut PIM): www.pim.wiwi.uni-due.de
 Website (Projekt OrGoLo):
 http://www.orgolo.wiwi.uni-due.de/

ISSN: 1866-9255



Das Verbundprojekt Organisatorische Innovationen mit Good Governance in Logistik-Netzwerken (OrGoLo) wird im Rahmen des Spitzenclusters „EffizienzCluster LogistikRuhr“ mit Finanzmitteln des deutschen Bundesministeriums für Bildung und Forschung (BMBF) gefördert (Förderkennzeichen: 01IC10L20A) und vom Projektträger im Deutschen Zentrum für Luft- und Raumfahrt e.V. (DLR) – Softwaresysteme und Wissenstechnologien (PT-SW) begleitet. Die Projektpartner danken für die großzügige Unterstützung ihrer Forschungs- und Transferarbeiten.

Partner des Verbundprojekts:

- admoVa Consulting GmbH
- bdf consultants GmbH
- DST – Entwicklungszentrum für Schiffstechnik und Transportsysteme e.V.
- Duisburger Hafen AG
- Lufapak GmbH
- relamedia GmbH
- SimulationsDienstleistungsZentrum SDZ GmbH
- TraffGo HT GmbH
- Universität Duisburg-Essen, Institut für Produktion und Industrielles Informationsmanagement
- Universität Duisburg-Essen, Lehrstuhl für Allgemeine Betriebswirtschaftslehre und Operations Management
- Universität Duisburg-Essen, Lehrstuhl für Transportsysteme und -logistik – Professur für Technische Logistik
- w3logistics AG



Universität Duisburg-Essen – Campus Essen
Institut für Produktion und Industrielles Informationsmanagement

Projektberichte des Verbundprojekts OrGoLo

ISSN 1866-9255

- Nr.1 Zelewski, S.: Überblick über das Verbundprojekt OrGoLo – Organisatorische Innovationen mit Good Governance in Logistik-Netzwerken. Essen 2011.
- Nr. 2 Kowalski, M.: Lastenheft für ein prototypisches Software-Tool zur Unterstützung des Case-based Reasonings (CBR-Tool). Essen 2011.
- Nr. 3 Robles, M.: Technische Rahmenbedingungen zur Gestaltung globaler Logistiknetzwerke. Essen 2011.
- Nr. 4 Lautenschläger, H.: Gestaltung globaler Logistiknetzwerke mit dezentralen Kompetenzen. Essen. 2011.
- Nr. 5 Lautenschläger, H.: Innovative Instrumente zur Gestaltung globaler Logistiknetzwerke. Essen 2011.
- Nr. 6 Lautenschläger, M.: Rechtliche Rahmenbedingungen für die Gestaltung globaler Logistiknetze. Essen 2011.
- Nr. 7 Leisten, R.: Analyse wirtschaftlicher Rahmenbedingungen zur Gestaltung globaler Logistiknetzwerke. Essen 2011
- Nr. 8 Kowalski, M.; Kovacevic, H.: State-of-the-art von CBR-Tools. Essen 2011.
- Nr. 9 Kowalski, M.; Kater, D.: Case-based Reasoning in Supply Chains – Qualitatives Case Retrieval. Essen 2011.