



UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

Universität Paderborn

Fakultät für Elektrotechnik, Informatik und Mathematik

Institut für Informatik

Prof. Dr. Gerd Szwillus

Wintersemester 2008 / 2009

Bearbeitungszeitraum:

20.01.2009 – 20.07.2009

Diplomarbeit

Prototypische Implementierung eines semantischen Assistenten für eine kollaborative Lernplattform

vorgelegt von:

Michael Düpjohann

Meerwiesenstr. 29

33442 Herzebrock-Clarholz

Telefon: +49 (0)177 - 61 31 51 4

E-Mail: m.duepjohann@gmx.net

Fakultät: Elektrotechnik, Informatik und

Mathematik

10. Fachsemester

Matrikelnummer: 6287050

ABSTRACT

Gegenwärtig ist die meist genutzte und effektivste Möglichkeit der Informationsbeschaffung im Internet die zuweilen langwierige Recherche mit verschiedenen Suchmaschinen. Dabei sollte überlegt werden, ob diese Form der Informationssuche auch in Zukunft die einzige Option bleiben soll. Im Zuge des Web 2.0 hat sich herausgestellt, dass viele Benutzer bereit sind, sich im Internet aktiv zu beteiligen. Dies geschieht unter anderem in Form von Annotationen von Datenobjekten wie Texten, Bildern und Videos. Diese Auszeichnungen helfen die Objekte miteinander zu verknüpfen. Das vom W3C entwickelte Semantic Web verfolgt ebenso dieses Ziel. Basierend auf Ontologien werden Inhalte verknüpft und können so ausgegeben werden, dass Maschinen in der Lage sind diese Informationen auszuwerten und in Bezug zu setzen. In dieser Arbeit soll untersucht werden, wie sich unter Einbeziehung von Aspekten des Web 2.0 und semantischen Techniken ein Empfehlungssystem entwickeln lässt, welches Benutzern das Finden für sie interessanter Inhalte erleichtert.

Schlagwörter: Semantic Web / RDF / microformats / Web 3.0 / Tagging / Empfehlungssystem

Inhaltsverzeichnis

1	EINLEITUNG	1
1.1	Information overload im Wissensmanagement?	3
1.2	Das Semantic Web als technischer Hoffnungsträger	4
1.3	Zielsetzung und Aufbau der Arbeit	5
2	WISSENSMANAGEMENT IN KOLLABORATIVEN LERNPLATTFORMEN	7
2.1	Unstrukturiertes Wissen als technische Herausforderung	7
2.2	Intelligente Wissensverknüpfung als vielversprechende Lösung	8
3	DAS SEMANTIC WEB AUS INFORMATIONSTECHNISCHER SICHT	9
3.1	Die Bedeutung von Semantik aus Sicht der Informationstechnologie	11
3.2	Entwicklungsformen / Ausprägungen	13
3.2.1	Insellösungen	15
3.2.1.1	Tagging	15
3.2.1.2	Automatische Semanticerstellung	18
3.2.1.3	Ausblick	20
3.2.2	Semantic Web	21
3.2.2.1	Microformats	21
3.2.2.2	W3C Standards	25
3.2.3	Sicherheit	36
4	TECHNISCHE UMSETZUNG	38
4.1	Vokabulare	38
4.1.1	Dublin Core – semantische Grunddateninformation	40
4.1.2	SKOS – Beschreibung von Wissensressourcen	41
4.1.3	Tag – Beschreibung von Tags	42
4.1.4	SCOT – erweiterte Tag und Tag Cloud Beschreibung	43
4.2	Werkzeuge	48
4.2.1	Zend Framework	48
4.2.2	ARC2	48

4.2.3	Ontologieeditoren	49
4.3	Vokabularmanager.....	51
4.3.1	Anforderungsdefinitionen.....	51
4.3.2	Umsetzung Vokabularmanager	51
4.4	Empfehlungssystem	52
4.4.1	Anforderungsdefinitionen.....	52
4.4.2	Komponenten.....	53
4.4.3	Funktionsbeschreibungen	55
4.4.4	Empfehlungsalgorithmus	57
4.4.5	Vergleich zu anderen Empfehlungssystemen	64
4.4.6	Ausblick	65
5	ZUSAMMENFASSUNG UND AUSBLICK	67
5.1	Zusammenfassende Betrachtung der Ergebnisse.....	67
5.2	Ausblick	67
ANHANG A		69
ANHANG B		75
GLOSSAR		76
LITERATURVERZEICHNIS.....		82
INTERNETQUELLENVERZEICHNIS.....		84
EIDESSTATTLICHE ERKLÄRUNG		88

ABBILDUNGSVERZEICHNIS

Abb. 1: Gliederung und Aufbau der Arbeit	6
Abb. 2: Das Portal learnuzz	8
Abb. 3: Semantische Treppe	12
Abb. 4: Struktur und Zusammenhang der theoretischen Modelle	14
Abb. 5: Suchergebnis von delicious.....	16
Abb. 6: GRDDL im Einsatz.....	34
Abb. 7: Technologien für das semantic Web	35
Abb. 8: Verbindung von SCOT zu anderen Vokabularen.....	40
Abb. 9: Die Tag Ontologie als Graph	43
Abb. 10: Graphische Darstellung der SCOT Ontologie	43
Abb. 11: Schematische Darstellung der SCOT Tag Cloud mit ihren Eigenschaften.....	44
Abb. 12: Schematische Darstellung eines SCOT Tag mit seinen Eigenschaften.....	45
Abb. 13: Schemantische Darstellung der Zusammenhangstärke zwischen Tags in SCOT.....	46
Abb. 14: Graphische Darstellung der SCOT Ontologie	47
Abb. 15: Ansicht einer Ontologie im Protege Editor	50
Abb. 16: Der Vokabularmanager	51
Abb. 17: Ausgabe von Empfehlungen zu einer Gruppe	57
Abb. 18: Objekte eines Tags sammeln	58
Abb. 19: Sammlung der Zusammenhangs-Tags	59
Abb. 20: Minimale und maximale Oberklassen gesammelt.....	60
Abb. 21: Ermitteln der zum Tag gehörenden Objekte	61
Abb. 22: Objekte in Ausgabeliste einfügen	62

TABELLENVERZEICHNIS

Tabelle 1: Einfacher RDF-Graph	26
Tabelle 2: Missverständnisse in binären Graphen	28
Tabelle 3: RDFS Klassen	29
Tabelle 4: RDFS Zusammenhänge für Prädikate	29
Tabelle 5: Restriktionen durch RDFS definieren.....	29
Tabelle 6: Unterschiede in den OWL Versionen	30
Tabelle 7: Häufig genutzte Dublin Core Auszeichnungen	41
Tabelle 8: Eigenschaften eines Tags in der Ontologie.....	53
Tabelle 9: Eigenschaften einer Tag Cloud in der Ontologie	54

ANHANGSVERZEICHNIS

ANHANG A	69
A.1 Richard Newman's Tag Ontology	69
A.2 SCOT – Übersicht.....	70
A.3 SCOT – Tag Cloud	71
A.4 SCOT – Tag.....	72
A.5 SCOT – Cooccurrence	73
A.6 SCOT – Tagging Activity.....	74
ANHANG B	75
B.1 STUDENTSERVICE GmbH.....	75

ABKÜRZUNGSVERZEICHNIS

Abb.	Abbildung
API	Application Programming Interface
ARPA	Advanced Research Projects Agency
B2B	Business to Business
B2C	Business to Consumer
CMS	Content Management System
CSS	Cascading Style Sheet
DTD	Document Type Definition
E-	Electronic-
GRDDL	Gleaning Resource Descriptions from Dialects of Languages
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
OWL	Web Ontology Language
PDF	Portable Document Format
RFC	Request for Comment
RDF	Resource Description Format
RDFS	RDF Schema
RSS	Real Simple Syndication
SGML	Standard Generalized Markup Language
SPARQL	SPARQL Protocol and RDF Query Language
SQL	Structured English Query Language
SWRL	Semantic Web Rule Language
W3C	World Wide Web Consortium
WWW	World Wide Web
XML	eXtensible Markup Language
XHTML	Extensible HyperText Markup Language
XSLT	eXtensible Stylesheet Language Transformation
µf	microformats

1 EINLEITUNG

*“The Semantic Web is a web of data,
in some ways like a global database.”¹*

Das Internet – unendliche Weiten. Fast alles ist verfügbar, es muss nur gefunden werden. Waren die Informationen im World Wide Web in den Anfängen sehr überschaubar, ist es dem Benutzer heute kaum noch möglich ohne maschinelle Unterstützung an die gewünschten Informationen zu gelangen. Doch auch diese Technik stößt schnell an ihre Grenzen und bedarf einer aktiven Beteiligung des Nutzers in Form von Recherchen mit Suchmaschinen oder ähnlichem. Denn der Mensch ist der einzige, der die Informationen richtig verstehen und deuten kann, maschinell können sie nur verarbeitet werden. Dabei wäre es für den Benutzer wünschenswert, wenn ein maschineller Assistent seinen Informationsbedarf erkennt und ihm dementsprechend Empfehlungen bereitstellt. Für diese Herausforderung benötigt es allerdings Techniken und Standards, die es der maschinellen Verarbeitung erlauben die Informationen nicht nur zu verarbeiten, sondern auch zu verstehen, zu interpretieren und die Zusammenhänge zu erkennen. Diesem Ziel widmet sich das Semantic Web in verschiedenen Ansätzen und Umsetzungen.

Der Beginn des Internetzeitalters

Bis die technischen Möglichkeiten für eine solche Entwicklung entstanden waren, bedurfte es eines langen Weges. Während des kalten Krieges wurde 1957 vom amerikanischen Verteidigungsministerium das ARPA (Advanced Research Projects Agency) gegründet. Aus dieser Organisation leitet sich auch das erste Internet namens ARPAnet ab. Die Idee Rechner miteinander zu verbinden entstand aus der Intention, dass die Menschen in einem Projekt nicht autoritär geführt und in Konkurrenz handelnd, sondern zusammen arbeiten sollten. Dies sollte im besten Fall nicht, wie es schon häufig geschah, an nur einem Großrechner stattfinden, sondern verteilt an verschiedenen Terminals. Es entstand der Entwurf der RFC (Request for Comment)-Philosophie. In dieser diskutieren viele verschiedene Menschen gemeinsam über ein Thema, wie es heute noch immer in Mailinglisten geschieht, wobei die Kommentare einer Person in der anfänglichen Zeit auf einen Satz beschränkt waren. Ebenfalls sollten die Ressourcen von Großrechnern durch Ressourcensharing besser genutzt werden. Im Jahr 1969 wurden die ersten Rechner miteinander verbunden und es entstand der Vorläufer des Internets, das ARPAnet. Es wird behauptet, dass dieses Netz entworfen wurde, um im Fall eines Nuklearschlages zu überleben.² Diese Aussage ist allerdings sehr umstritten. Zweifelsfrei ist, dass aus einem Programm für die Benutzerkommunikation auf Großrechnern 1972 das erste Mailprogramm entwickelt wurde, welches schon bald 75% des Datenvolumens ausmachte. Zu dieser Zeit umfasste das Netz 15 Großrechner. Im Jahr 1989 ging das für Forschungs- und Militärzwecke gegründete ARPAnet in das 1985 gegründete zivile NSFnet (National

¹ Vgl. Berners-Lee (1998).

² Vgl. Hardy (1993).

Science Foundation Net) über, welches zu dieser Zeit schon aus 100.000 Hosts bestand. Damit es mit anderen Netzen verbunden werden konnte, wurde 1983 das TCP (Transmission Control Protokoll) Internet Protokoll eingeführt.

Im Forschungszentrum CERN³, in dem Informationsorganisation eines der größten Probleme war, wurde im Jahr 1989 von Tim Berners-Lee⁴ die Hyper Text Markup Language (HTML) entwickelt. Sie entstand aus dem schon existierenden SGML (Standard Generalized Markup Language von 1986), bot darüber hinaus aber die Möglichkeit, Datensätze und Grafiken ins Dokument einzubauen. Für die Wissensorganisation ein entscheidender Vorteil. Den Durchbruch brachten im Jahr 1993 der Browser Mosaic⁵ beziehungsweise 1994 Netscape Navigator.⁶ Ein Jahr später wurde das Internet von der amerikanischen Regierung vor staatlichen Einflüssen dereguliert und dem freien Markt überlassen. Damit begann, nach der im Wesentlichen militärischen und der universitären Nutzung, die Massennutzung. Das Internet wurde populär, der Internetboom setzte ein und mit ihm die Kommerzialisierung. Obwohl es schwierig war, monetäre Umsätze im Internet zu generieren, stürzten sich viele Firmen auf die neue Technik. Dieses Wachstum wurde allerdings im Jahr 2001 durch den Zusammenbruch der neuen Märkte gebremst.⁷

Bis zu diesem Zeitpunkt wurden Webseiten und Inhalte von Experten eingestellt, die wesentliche Einnahmequelle war allerdings Werbung in Form von Text und Bildeinblendungen. Damit hatten sich viele neue Unternehmen übernommen und die Zahlungsbereitschaft der Benutzer überschätzt. Profitiert haben davon größtenteils die Suchmaschinen, welche sich mit Werbung sehr gut finanzieren konnten. Nach dem Zusammenbruch im Jahr 2001 gingen viele neue Firmen insolvent, es überlebten aber auch einige heute bekannte, wie diverse Online-Shops.⁸ Es reichte nicht mehr, eine gute Idee zu haben, sie musste auch wirtschaftlich tragbar sein. Zu dieser Zeit hatte Internet bereits eine bedeutende Größe erreicht, aber die Anzahl an Privatleuten, die eigene Inhalte meist in Form einer eigenen Homepage online stellten, war noch sehr gering, denn es bedurfte noch spezifischer Fachkenntnis.⁹ Daher waren es nur Webmaster oder technisch interessierte Anwender, die sich mit einer eigenen Homepage ins Internet wagten. Wenn auf die Anfänge und die Intention zurückgeschaut wird, ist dies eigentlich sehr verwunderlich, denn das Internet sollte eine Plattform darstellen, in der jeder Beteiligte sein Wissen teilen kann. Doch im Jahr 2005 wandelte sich das Web: Zum einen wurde die technische Ausstattung der einzelnen Benutzer besser, denn die Computer wurden ebenso wie die Internetanschlüsse schneller und die Anzahl solcher Anschlüsse stieg stetig. Zum anderen entstanden die entscheidenden Tools wie Online Shops, Content Management Systeme und vieles mehr. Ab dieser Zeit konnte jeder seine Ware bei ebay¹⁰ oder amazon.de¹¹ verkaufen, seinen

³ Europäische Organisation für Kernforschung, siehe <http://www.cern.de/>, letzter Zugriff am 16.07.2009.

⁴ <http://www.w3.org/People/Berners-Lee/>, letzter Zugriff am 16.07.2009.

⁵ <ftp://ftp.ncsa.uiuc.edu/Mosaic/>, letzter Zugriff am 16.07.2009.

⁶ <http://browser.netscape.com/>, letzter Zugriff am 16.07.2009.

⁷ Vgl. Kuhn (2007).

⁸ Siehe amazon.de oder ebay, Tenbrock (2000).

⁹ Vgl. Blumauer/ Pellegrini (2009), S 13ff.

¹⁰ <http://www.ebay.de/>, letzter Zugriff am 16.07.2009.

¹¹ <http://www.amazon.de/>, letzter Zugriff am 16.07.2009.

eigenen Blog¹² aufsetzen oder sich unkompliziert eine Profilseite bei XING¹³ oder facebook¹⁴ anlegen. Ebenfalls entstanden die Wikis, bei denen jeder Benutzer sein Wissen mit anderen teilen konnte. Von der technischen Seite waren solche Systeme schon eher möglich, doch jetzt setzten sie sich in der breiten Masse der Bevölkerung durch. Vorher wurde von einem dokumentenbasierten Web gesprochen, dem Web 1.0. Dort waren die Inhalte zumeist statische Seiten oder Downloads, welche von den vorhin genannten Webmastern angelegt wurden. Dieses Muster wurde durchbrochen und es entstand ein Web der Dienste, in dem jeder auf einfache Art und Weise mitmachen konnte. Erst später kam der Begriff Web 2.0¹⁵ für diesen Umbruch auf, obwohl er sehr unscharf und im strengen Sinne falsch ist, denn die Versionsnummer suggeriert einen kompletten Systemwechsel. Es handelt sich jedoch nur um eine Weiterentwicklung, welche parallel entstanden ist. Dennoch galt dieser Name als Symbol für eine neue Zeit.¹⁶

1.1 Information overload im Wissensmanagement?

Auch wenn das Web 2.0 als Revolution galt, war es eine Ergänzung zum bestehenden Netz und kein Ersatz oder eine grundlegende Innovation. Allerdings schuf dieses neue Mitmach-Internet weitere ungeahnte Möglichkeiten. Es bestand nun erstmals die Chance, das Wissen der gesamten Menschheit zu sammeln. Haben zuvor nur Experten ihr Wissen auf so breiter Basis verteilt, konnte nun jeder seinen Beitrag dazu geben, so dass kooperative Wissensstrukturen entstanden. Bekanntestes Beispiel dafür ist das Onlinelexikon Wikipedia¹⁷. In einem solchen Wiki¹⁸ kann jeder Anwender sein Wissen den anderen Benutzern mitteilen, so dass schnell eine große Menge an Wissen gesammelt werden kann. Auch die Bekannten traditionsreichen Wissenssammlungen wie die Lexikareihe Brockhaus¹⁹ konnten sich dem Onlinedrang nicht entziehen, obwohl dieses Wissen nur kostenpflichtig einsehbar ist, da es noch immer von Experten aktualisiert wird. An diesem und anderen Beispielen²⁰ wird deutlich, dass das gesammelte freie Wissen der Bevölkerung dem expliziten Wissen der Experten ebenbürtig, wenn nicht sogar überlegen ist. Als eine weitere Möglichkeit der Wissensweitergabe ist der Trend Blogszuführen. Das Wort Blog setzt sich aus den Begriffen Web und Log zusammen und ist eine Art Onlinetagebuch oder Journal. Viele Menschen, egal ob Laien oder Experten, schreiben über für sie interessante Themen, welche sehr häufig auch Inhalte umfassen, mit denen sie häufig zu tun haben und der Autor deshalb entsprechendes Wissen mitbringt.²¹ Ein Rückblick an die Anfangszeit des Internets zeigt, dass es genau das ist, wovon die Erfinder geträumt haben. Jede Firma kann heute ihren eigenen Blog oder ihr eigenes Wiki führen,

¹² Vgl. Alby (2007), S. 21.

¹³ <https://www.xing.com/>, letzter Zugriff am 16.07.2009.

¹⁴ <http://www.facebook.com/home.php>, letzter Zugriff am 16.07.2009.

¹⁵ Vgl. Alby (2007), S. 1.

¹⁶ Vgl. Kampmann (2001).

¹⁷ <http://de.wikipedia.org/>, letzter Zugriff am 16.07.2009.

¹⁸ Ebersbach/ Glaser/ et al. (2008), S. 11.

¹⁹ <http://www.brockhaus.de/nachschlagen/>, letzter Zugriff am 16.07.2009.

²⁰ Vgl. Stern (2004).

²¹ Vgl. Schütt (2006).

in dem die Mitarbeiter größtenteils gerne und freiwillig etwas hinzufügen. Auch gibt es zu vielen verschiedenen Themen ein Wiki, in dem die speziellen Informationen zu dem Thema gesammelt werden. Im privaten Bereich sind sehr viele Blogs entstanden, denn jeder schreibt gerne über Dinge die ihn interessieren. Dies alles lässt sich zudem per Atom oder RSS²² abonnieren, so dass ein Benutzer sich seine eigene persönliche Sammlung von Nachrichten zusammenstellen kann. Die Anzahl der Blogs ist mittlerweile so groß, dass dafür spezialisierte Suchmaschinen wie Technorati²³ oder Google Blogs²⁴ existieren. Auch die Anzahl der Wikis ist zu einer nicht überschaubaren Menge heran gewachsen, obwohl es hier einige große Systeme gibt, die sich von der Masse abheben. Dennoch führt der Wandel des Internets von einem Expertensystem zu einem Massensystem zu einer Flut von Informationen. Nicht umsonst haben Suchmaschinen im Internet einen derart großen Stellenwert. Gab es früher noch Bücher, in denen die wichtigsten Internetadressen gelistet waren, ist dies bei der heutigen Masse und der Schnelligkeit, mit der Informationen geändert werden, nicht mehr möglich. Doch jeder, der bereits im Internet für bestimmte Themen recherchieren musste, weiß, dass es alles andere als einfach ist die benötigten Informationen zu finden. Sie sind vorhanden, ziemlich sicher, die Frage ist nur: Wo?

Das Problem besteht also darin in diesem dezentralen und sehr heterogen aufgebauten Internet die passenden Informationen ausfindig zu machen und gegebenenfalls mit Informationen aus anderen Quellen zu verbinden. Für den Menschen ist es kein Problem die Inhalte einer Seite, eines Bildes oder eines Videos zu verstehen und zu interpretieren. Für die heutigen Suchmaschinen stellt dies hingegen eine sehr große Herausforderung dar. Die Technik beruht zum größten Teil darauf, bestimmte Texte einer Seite mit den eingegebenen Suchworten zu vergleichen. Die Bedeutung der Information oder in welchem Zusammenhang diese Information auftritt, ist für die Suchmaschine nicht erkennbar. Daher können Informationen auch nur schwer automatisch miteinander in Verbindung gesetzt werden.²⁵

1.2 Das Semantic Web als technischer Hoffnungsträger

Die Herausforderung besteht deshalb darin, eine künstliche Intelligenz zu entwickeln, welche die Funktion des Menschen nachahmt und Informationen interpretieren kann. Bei dem heterogenen Aufbau des Internets ist dies aber in absehbarer Zeit noch nicht umsetzbar. Ein erster Schritt wäre ein System umzusetzen, durch das die Maschine die Informationen in einem Dokument erkennt und versteht. Wobei ein menschliches Verständnis zurzeit noch illusorisch ist. Es sollte einen Weg geben, mit dem die Maschine mit den Informationen umgehen kann, sie beispielsweise in Orte und Personen klassifiziert und für den Menschen passend aufbereitet. Genau dies ist das Ziel des Semantic Web. Wie der Erfinder des Internets Tim Berners Lee es in einem Interview sagte: *„Die Informationen, wer meine Freunde sind und wo sie sich gerade befinden, sollte man in Verbindung*

²² Alby (2007), S. 137ff.

²³ <http://technorati.com/>, letzter Zugriff am 16.07.2009.

²⁴ <http://blogsearch.google.de>, letzter Zugriff am 16.07.2009.

²⁵ Vgl. Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 10ff.

setzen mit den Orten, wo ich gerne Kaffee trinken gehe. Die Informationen wer meine Freunde sind und wo sie sich gerade befinden können im Internet vorhanden sein. Auch die Orte, wo man gut Kaffee trinken kann.“ (Berners-Lee (2007)) Der erste Schritt ist folglich dem Computer beizubringen, diese Informationen als das zu erkennen was sie sind, Objekte mit einer bestimmten Bedeutung. Im zweiten Schritt ist die Herausforderung diese Informationen intelligent miteinander zu verknüpfen. Dies mag für den Benutzer nach künstlicher Intelligenz aussehen, aber alles was die Maschine dazu benötigt sind Meta-Informationen zu dem bestehenden „Web of Data“. Eine Information an sich ist relativ bedeutungslos, wenn sie nicht gefunden und interpretiert werden kann. Es benötigt dementsprechend einen offenen und einheitlichen Standard um Informationen im Internet mit maschinenverstehbaren Daten zu untermauern. Das World Wide Web Consortium (W3C) entwickelt und definiert die weltweiten Standards für interoperable Technologien wie das Internet. Für genau dieses Problem hat das World Wide Web Consortium²⁶ mit ihren Standards XML, RDF(S) und OWL eine Lösung erstellt. Dieses Problem wurde schon Ende der neunziger Jahre erkannt und zu lösen versucht, lange bevor es in der breiten Masse wahrgenommen wurde. Es ist demzufolge kein neues Phänomen, sondern eine stetige parallele Entwicklung, die sich allerdings noch nicht vergleichbar erfolgreich durchsetzen konnte wie es das Web 2.0 seinerzeit getan hat. Mit den genannten Techniken wird es möglich Wissen in so genannten Ontologien zu modellieren. Wie dies genau geschieht wird in den nächsten Kapiteln erklärt. Das Resultat sind Daten, die mit zusätzlichen Informationen für den Computer angereichert wurden und auf einfache Weise automatisch ausgelesen werden können. Eine Ontologie sorgt dabei für eine wohl geformte Struktur, so dass die Informationen in Verbindung zueinander gestellt werden können.

1.3 Zielsetzung und Aufbau der Arbeit

Generell soll dem Benutzer geholfen werden, die für ihn interessanten Objekte und Informationen im Internet zu finden. Um dies in einer realistisch umsetzbaren Größe zu versuchen, beschränkt sich diese Arbeit auf das kollaborative Lernportal learnuzz, welches durch die **STUDENTSERVICE** GmbH aus Paderborn bereitgestellt wird.²⁷ Dieses Portal für Studierende und Auszubildende ermöglicht es dem Benutzer Multiple-Choice²⁸ Fragen zu Klausuren einzustellen und zu beantworten. Zudem existieren Netzwerkfunktionalitäten wie Profilseiten oder Gruppen. Das Ziel dieser Arbeit ist es, dem Benutzer bei seiner Navigation durch dieses Lernportal auf weitere interessante Inhalte hinzuweisen. Dies geschieht mit Hilfe eines semantischen Assistenten. Der Assistent soll anhand der Umgebung, wie zum Beispiel der aktuellen Seite oder durch Berücksichtigung von getätigten Benutzeraktionen Empfehlungen generieren. Diese können sich auf andere Fragen, Gruppen oder sonstige Objekte des Lernportals beziehen. Somit können sowohl das Profil des Benutzers mit für ihn interessanten Objekten verknüpft werden, aber auch die Objekte unter sich.

²⁶ www.w3.org, letzter Zugriff am 16.07.2009.

²⁷ <http://www.learnuzz.net>, letzter Zugriff am 16.07.2009, weitere Informationen zur learnuzz.net Plattform und zur **STUDENTSERVICE** GmbH befinden sich im Anhang B.1.

²⁸ Darunter versteht man Fragen, zu denen es eine feste Auswahl von Antwortmöglichkeiten gibt, vgl. Jacobs, B. (2005).

Abb. 1: Gliederung und Aufbau der Arbeit²⁹

Die Arbeit ist in fünf Kapitel unterteilt. Das erste Kapitel enthält eine allgemeine Einführung in das Thema. Es wird die Entstehung und Entwicklung der Technologien und deren Umgebung erklärt, welche zu der Problemstellung dieser Diplomarbeit, die Notwendigkeit eines semantischen Assistenten, führen. Im zweiten Kapitel wird untersucht, ob bestimmte Lösungsansätze bereits existieren. Dazu werden die verschiedensten Seiten, welche im Internet verfügbar sind nach Existenz solcher Assistenten untersucht. Bereits vorhandene Umsetzungen werden getestet und deren Funktionalität auf Praxistauglichkeit untersucht. Die möglichen Erfahrungen aus den Systemen werden gesammelt um so eine Schnittmenge an guten Lösungen zu erhalten. Im dritten Kapitel wird die theoretische Herangehensweise fokussiert. Es wird untersucht, ob es, neben den im vorherigen Kapitel eventuell auf reine Praxis angepassten Lösungen, theoretische Überlegungen gibt, welche in ihrem Umfang wesentlich größere Kreise ziehen. Die bedeutungsvollen Themen sind dabei RDF(S), OWL sowie microformats. Diese Erkenntnisse werden erklärt, ausgewertet und nach dem Nutzen für unsere Bedürfnisse bewertet. Nachdem mögliche Lösungsansätze im Internet untersucht wurden und die theoretische Herangehensweise an dieses Thema aufgegriffen wurde, wird im vierten Kapitel auf diesen Erkenntnissen beruhend eine eigene Umsetzung entwickelt. Diese muss, den

²⁹ Quelle: Eigene Darstellung.

Zielen der Arbeit entsprechend, dem Benutzer das Finden der für ihn interessanten Inhalte erleichtern. Neben der praktischen Umsetzung wird hier die Herangehensweise mit den verschiedenen benötigten Softwarewerkzeugen sowie deren Nutzen und Einsatz beschrieben. Ebenso werden für die Umsetzung relevante Vokabulare untersucht und erläutert. Der Bedarf von Eigenentwicklungen wird anhand der Systemanforderungen erklärt und deren Funktionalität detailliert aufgelistet. Zudem wird gezeigt, auf welchen theoretischen Überlegungen oder schon existenten Lösungen diese Entwicklungsansätze beruhen. Zum Schluss werden mögliche Vor- und Nachteile diskutiert sowie Potentiale aufgelistet. Im letzten Kapitel wird das Erreichte in den allgemeinen Kontext gebracht, zusammengefasst und Probleme sowie gewonnene Erkenntnisse über zukünftige mögliche Entwicklungen und deren Potentiale aufgegriffen.

2 WISSENSMANAGEMENT IN KOLLABORATIVEN LERNPLATTFORMEN

Bevor mit der Analyse verschiedener Systeme begonnen wird, sollte zuerst geklärt werden was im Allgemeinen unter einer kollaborativen Lernplattform zu verstehen ist. Zum einen ist die Anforderung, die Plattform kollaborativ zu gestalten. Dieses Wort leitet sich von Kollaboration (lateinisch *co* = „zusammen“, *laborare* = „arbeiten“) ³⁰ ab und bedeutet soviel wie Zusammenarbeiten. Der Anwender befindet sich also auf einer Plattform auf der zusammen etwas erarbeitet oder bearbeitet wird. Dies entspricht dem Gedanken des Web 2.0, bei dem jeder Benutzer dazu aufgerufen ist seinen Beitrag auf der Webseite zu erstellen. Kollaboratives Arbeiten ist nicht neu, insbesondere in Unternehmen wird dies schon seit längerem verstärkt von Mitarbeitern umgesetzt. Jedoch ist die Möglichkeit dies mit unbekannten Menschen im Internet zu verwirklichen eine Technik, die sich erst in den letzten Jahren verstärkt durchsetzt. Nicht notwendig, aber durchaus hilfreich für eine gute Zusammenarbeit ist die Integration von Netzwerkfunktionalitäten. Darunter wird verstanden, dass den Benutzern die Möglichkeit geboten wird sich zu registrieren und eine Profilseite anzulegen. Dadurch rückt das System ab von der Anonymität und gibt den einzelnen Akteuren auf der Seite ein Gesicht. Weitere Funktionen, wie die Möglichkeit untereinander Nachrichten auszutauschen und Gruppen zu bilden, fördern ebenfalls den Zusammenhalt der Gemeinschaft und die Identifikation mit der Seite. Erfolgreiche Wikiseiten wie Wikipedia demonstrieren, dass dies zwar nicht unbedingt erforderlich ist, aber trotzdem ein durchaus erfolgreiches Konzept sein kann wie anhand der vielen sozialen Netzwerke im Internet sichtbar ist.

2.1 Unstrukturiertes Wissen als technische Herausforderung

Neben der Zusammenarbeit ist die Lernplattform der zweite Schwerpunkt des Portals. Dies bedeutet, dass die Webseite in irgendeiner Form das Lernen fördern soll. Die Plattform learnuzz hat sich zum Ziel gesetzt Schülern, Auszubildenden und Studierenden bei der Vorbereitung auf Klausuren zu helfen. Sie bietet die Möglichkeit Fragen zu stellen, welche danach gemeinschaftlich beantwortet werden können. Dabei wurde ein besonderer Fokus auf Multiple-Choice Fragen gelegt, welche

in vielen Prüfungen vorkommen. Jede Antwortmöglichkeit kann diskutiert und bewertet werden, so dass der Benutzer zum Schluss eine statistische Auswertung über die besten Antwortmöglichkeiten erhält. Die Zusammenarbeit wird außerdem durch grundlegende Netzwerkfunktionalitäten wie Profilseiten ergänzt. In Entwicklung befinden sich außerdem ein System zur Bildung von Gruppen als auch ein Downloadportal für Dateien verschiedenster Art.

Das Einstellen und Beantworten von Fragen birgt allerdings Herausforderungen. Die Menge an Fragen und damit generiertes Wissen landet unstrukturiert in einem großen Pool und ist ab einer bestimmten Größe nur noch schwer zu erschließen. Ein Benutzer kann nicht überblicken, ob seine Frage oder sein Download schon vorhanden ist. Wenn dieser sich gerade eine Frage, Download oder Gruppe anschaut, ist ihm nicht bekannt, ob es noch weitere, ähnliche und damit ebenfalls für ihn interessante Objekte existieren. Sobald die Menge an Wissen und Inhalt für den einzelnen Nutzer nur noch schwer oder mit erheblichem Aufwand zu überschauen ist, werden Automatismen benötigt, welche dem Benutzer das Finden von Inhalten erleichtern.

The screenshot displays the learnuzz website interface. At the top left is the logo 'learnuzz' with the tagline 'study . together . mobile'. To the right is a login section with a 'Hilfe' link, a language dropdown set to 'Deutsch', and input fields for 'E-Mail' and 'Passwort', followed by a 'Login' button. Below the login section is a registration section titled 'Jetzt registrieren!' with the text 'learnuzz ist kostenfrei und jeder kann mitmachen'. It includes input fields for 'Vollständiger Name', 'Geschlecht', 'E-Mail', 'Passwort', and 'Status', followed by a 'Registrieren' button. A small disclaimer at the bottom of the registration section states: 'Mit der Registrierung bestätigst du, dass du unsere Spielregeln und Richtlinien zum Datenschutz gelesen hast und diesen zustimmst.' On the left side, there are three promotional cards: 'study' (Mit learnuzz lernst du ganz gezielt für deine Prüfungen und bist besser vorbereitet. Spare Zeit!), 'together' (Zusammen können wir viel mehr erreichen. Lerne mit Bekannten und Freunden. Sei dabei!), and 'mobile' (Nutze die Möglichkeiten des Internets. Lerne wo und wann immer du willst. Werde effektiver!). At the bottom left, there is a 'Frage stellen' section with the text 'Stell eine Frage ein und sende sie an deine Freunde!' and a link 'Jetzt eine Frage stellen!' next to a large question mark icon.

Abb. 2: Das Portal learnuzz³¹

2.2 Intelligente Wissensverknüpfung als vielversprechende Lösung

Dies können einerseits Suchmaschinen sein, in denen der Benutzer aktiv nach Inhalten suchen kann. Andererseits ist es wesentlich benutzerfreundlicher, wenn die Suche ebenfalls automatisch geschieht. Sie zeigt dem Benutzer ohne aktive Suche die für ihn interessanten Objekte an. Eine

³⁰ Vgl. Hildenbrand/ Rothlauf/ Heinzl (2006).

³¹ Quelle: Eigene Darstellung.

solche Aufgabe würde ein Empfehlungssystem übernehmen. Aber auch ein maschinelles System benötigt Kriterien, anhand derer Objekte in Verbindung gesetzt werden können. Dies kann auf verschiedenen Wegen geschehen. Meistens werden die Objekte mit Metadaten hinterlegt, welche Informationen über das Objekt darstellen, die im Inhalt nicht direkt erwähnt werden müssen. Dabei kann es sich um eine Kategorisierung handeln oder um eine zusätzliche Information in Form eines Schlagwortes. Diese Informationen können in Beziehung gesetzt und miteinander verglichen werden um auf dadurch eine Verbindung zwischen den Objekten herzustellen oder eine Struktur in dem Pool mit Objekten zu bilden. Denkbar ist auch Benutzeraktionen auszuwerten, um daraus Bezüge herzustellen. Dabei können die Benutzer selbst auch als Objekte angesehen werden, zwischen denen Verbindungen hergestellt werden. Benutzeraktionen sind einerseits Bewegungen des Benutzers auf der Seite, als auch Eingaben in das System wie das Einstellen von Fragen oder das Hinzufügen von Antworten und Metadaten. Neben diesen oberflächlicheren Verknüpfungen gibt es auch noch die in der Einleitung erwähnte Möglichkeit der semantischen Beziehungen. Dabei werden die Objekte durch Ontologien untermauert, welche auf der Ebene der Metadaten Beziehungen und Strukturen herstellen können.

3 DAS SEMANTIC WEB AUS INFORMATIONSTECHNISCHER SICHT

Bevor geklärt wird wofür Semantic Web genau steht, sollte die Vorgeschichte betrachtet werden. Diese beginnt nicht im Internet, sondern in Bibliotheken und Archiven. Schon früher stand der Mensch vor der Herausforderung das gesammelte Wissen zu ordnen, um einen schnellen Zugriff darauf zu haben. Daraus entstanden verschiedene Wissensorganisationssysteme welche heute auch für digitale Inhalte genutzt werden. Um das Ziel, das Wiederfinden von Informationen oder auch das Navigieren durch diese, zu erreichen, werden die Dokumente analysiert um Metadaten herauszuziehen oder zu bilden. Denn „*Wissensorganisationssysteme sind strukturierte Darstellungen von Metadaten (Informationen über Informationen).*“ (Blumauer/ Pellegrini (2009), S. 128) Diese können in drei Arten eingeteilt werden:

- Zum einen gibt es beschreibende Metadaten, welche den Inhalt des Dokumentes widerspiegeln.
- Außerdem gibt es Metadaten welche direkte Informationen aus dem Objekt sind, zum Beispiel das Format.
- Zuletzt gibt es noch die verwaltenden Metadaten, welche das Dokument um zusätzliche Informationen wie Benutzungsrechte erweitern.

Diese verschiedenen Metadaten müssen in der Art aufbereitet sein, dass sie die Struktur der Sichtweise des Suchenden widerspiegeln und ihm dadurch das Finden erleichtern. Dafür gibt es verschiedene Ordnungstypen. Dazu zählen:

- Die alphabetische Auflistung, wie sie in Wörterbüchern zu finden ist
- Die Kategorisierung
- Die geographische Einordnung

- Die zeitliche Einordnung
- Die hierarchische Ordnung

Neben diesen einfachen und sehr bekannten Strukturen werden die Systeme durch den Einsatz von Relationstypen wesentlich komplexer. Es gibt die gleichbedeutenden Relationen, die hierarchischen und die assoziativen. Gleichbedeutend heißt, dass in Ordnungssystemen wie dem Alphabetischen und dem Chronologischen die Dokumente gleichwertig sind, auch wenn eines weiter oben in der Liste steht. Hierarchisch bedeutet, dass die Dokumente in einer strengen Ordnung festgehalten sind und dadurch ein unterschiedliches Gewicht besitzen. In einer assoziativen Relation werden die Objekte durch ihren Inhalt beziehungsweise durch die Metadaten miteinander verbunden. Die Komplexität von Relationstypen kommt auf, wenn mehrere dieser Elemente zusammen angewendet werden.

Daraus entstanden verschiedene bekannte Wissensorganisationssysteme: Das Einfachste ist die *Begriffsliste*, welches eine Sammlung von Objekt beschreibenden Bezeichnungen ist. In dieser Liste werden normalerweise gleichbedeutende Relationen verwendet, welche in der Praxis zum Beispiel als Wörterbücher bekannt sind. *Glossare* sind alphabetische Auflistungen von Begriffen, welche kurz erläutert werden. Sie werden gerne in Büchern oder Publikationen verwendet.³² *Stichwortsysteme* sind alphabetisch geordnete Begriffe, welche aus dem Inhalt des Dokuments stammen. Die ähnlich gearteten *Schlagwortsysteme* sind jedoch viel mächtiger, da sie sich nicht nur auf Begriffen des Dokuments beruhen. Sie bestehen aus Begriffen, welche das Dokument beschreiben und somit frei wählbar sind. *Folksonomies*³³ bestehen aus Schlagworten, welche im englischen als *Tag* bezeichnet werden. Diese können wiederum von Benutzern für Objekte wie Bilder, Videos oder textuelle Informationen vergeben werden. Da bestimmte Tags häufig vergeben werden kann so eine Gewichtung der einzelnen Begriffe vorgenommen werden. Eine Übersicht erfolgt mittels einer *Tag Cloud*, in der die Tags anhand ihrer Gewichtung in unterschiedlichen Größen dargestellt werden. Das besondere und strittige an Folkonomies ist, dass die Begriffe nicht von Experten sondern von den normalen Benutzern vergeben werden. Dies stellt daher eine starke Abgrenzung und Neuerung zu den bisherigen Systemen dar. Bei *Klassifikationen* und *Taxonomien* werden die Begriffe ihrer Bedeutung nach in Gruppen eingeteilt. Diesen können Oberbegriffe zugeordnet werden, wobei ein Begriff auch in mehreren Gruppen vorhanden sein kann. Dadurch entsteht eine polyhierarchische Struktur. Ein *Thesaurus* ist eine nach Normen geregelte Struktur. Die Begriffe werden assoziativ in Verbindung gesetzt, zum Beispiel unterschiedliche Schreibweisen eines Wortes. Zuletzt gibt es noch die *Ontologien*, welche sehr unterschiedlich eingesetzt werden. Insbesondere in der Informations-Technologie dienen sie dazu Wissen durch Klassen, Attribute und Relationen zu modellieren. Einen offiziellen Standard gibt es nicht, allerdings praktische Umsetzungen durch Thesauri oder Wörterbücher.

Zur Modellierung eines Wissenssystems gibt es zehn Phasen welche durchlaufen werden müssen: Zuerst muss der Einsatzbereich abgegrenzt werden. Die Fragen, welcher Kontext vorhanden ist,

³² Als Beispiel siehe Glossar dieser Arbeit.

³³ Vgl. Alby (2007), S. 111.

Vgl. Vander Val (2007).

welche Benutzergruppe das System hauptsächlich nutzen wird, welche Inhalte benutzt werden und welches Vokabular sinnvoll ist, sollten diskutiert werden. Als Nächstes werden Wörter aus den vorhandenen Quellen sowie dem Kontext oder dem Dokument gesammelt. Diese werden daraufhin kontrolliert, um Ungereimtheiten wie Singular-, Pluralformen oder ähnliches zu vermeiden. Danach werden sie in eine Struktur gebracht, indem Klassen gebildet und die Wörter diesen zugeordnet werden. Nach der Auswahl einer geeigneten Relation werden diese Klassen nun mit den beschriebenen Relationen verbunden. Im nächsten Schritt kann die Eigenschaften jeder Klasse definiert werden. Die genaue Vorgehensweise variiert dabei je nach Modellierungstool. Danach wird das Wissensorganisationssystem um die konkreten Ausprägungen der einzelnen Objekte ergänzt, wodurch das spezielle Wissen eines jeden Systems entsteht. Zum Schluss muss das erstellte System evaluiert und danach stetig gewartet werden. Mit der Zeit ändert sich der Blickwinkel auf die Wissensobjekte, weshalb eine Wartung auch notwendig ist, wenn keine neuen Inhalte hinzugefügt wurden.³⁴

3.1 Die Bedeutung von Semantik aus Sicht der Informationstechnologie

Bevor Möglichkeiten überlegt werden, wie das Wissen im Semantic Web modelliert werden kann, sollte zunächst geklärt werden was Semantik genau bedeutet und was genau unter den für die Informatik wichtigen Ontologien zu verstehen ist.

Die Semantik ist im Wesentlichen die Lehre der Bedeutung eines Wortes. Sie unterscheidet zwischen Sache und Bedeutung. Wenn die Wörter „Sonnenaufgang“ sowie „Sonnenuntergang“ betrachtet werden, beziehen sich beide auf den Stern Sonne und dessen Bewegung, bezeichnen demnach die gleiche Sache. Jedoch ist die Bedeutung eine andere, da einerseits der Morgen, andererseits der Abend beschrieben wird. Die Semantik bezieht die Außenwelt, also den Interpret des Objektes und den Kontext mit ein. Ein Dokument wird selten ohne Bezug zu anderen Dingen gefunden und sobald eine Person dieses Dokument liest, wird es interpretiert und bekommt eine eigene Bedeutung. Um die Bedeutung festzuhalten, welche nicht direkt im Dokument vorhanden ist, gibt es Metadaten. Dies sind Daten über Daten. Eine Maschine kann nur Zeichenketten erkennen und suchen, einem Menschen hingegen ist es möglich eine Zeichenkette als Datum zu interpretieren und mit diesem Datum eventuell sogar Ereignisse zu verbinden. Die hinzugefügten Metadaten werden in der Informatik mittels RDF strukturiert. Dieser allgemeine Standard, auf den im Kapitel „W3C Standards“ genauer eingegangen wird, garantiert, dass die Daten systemunabhängig ausgelesen werden können. Dies ist wichtig, da das Semantic Web auf Nachhaltigkeit ausgelegt ist, wodurch es sich wesentlich vom Web 2.0 unterscheidet. Das Web 2.0 hat ebenfalls eine große Änderung hervorgebracht, jedoch geschah dies unkontrolliert und sehr spontan. Semantic Web hingegen ist ein langer Prozess, der sich schon seit einigen Jahren in der Entwicklung befindet und langsam beginnt sich durchzusetzen. Es geht dabei nicht nur um die reine Wissensorganisation, welche sich auf das Navigieren und Wiederfinden von Inhalten beschränkt, sondern auch um die Möglichkeit der Automatisierung. Dadurch sollen Geschäftsprozesse im Business to Consumer (Unternehmen

34 Vgl. Blumauer/ Pellegrini (2009), S. 127ff.

zum Endkunden) und im Business to Business (Unternehmen zum Unternehmen) automatisiert werden. Diese Arbeit beschränkt sich allerdings auf den Teil des reinen Wissensmanagements. In Abgrenzung zum Semantic Web gibt es noch die semantischen Technologien, welche sich nicht nur auf das Internet beschränken, sondern bereits jetzt schon verschiedene Arbeitsprozesse unterstützen, welche unter der Informationsflut leiden.³⁵

Eng verbunden mit dem Semantic Web sind die Ontologien. Diese wurden insbesondere für den Bereich der Informatik entwickelt, um dort das Wissen zu strukturieren. Sie werden formal dargestellt und sollen ebenfalls systemunabhängig sein. Da sie sehr komplexe Gebilde sind, werden sie zumeist von Experten im Top Down Verfahren modelliert, dem sogenannten „Ontology Engineering“. Top Down bedeutet, dass vom Allgemeinen zum Speziellen vorgegangen wird. Im Gegensatz dazu steht der moderne Ansatz der Folksonomy, welche eine von vielen Benutzern erstellte Wissensbasis ist und Bottom Up modelliert wird, also bei den einzelnen Informationen zu strukturieren anfängt. Die Komplexität von Ontologien und damit deren Einordnung in die verschiedenen Modellierungen von Wissensorganisationen lässt sich anhand einer semantischen Treppe darstellen.³⁶

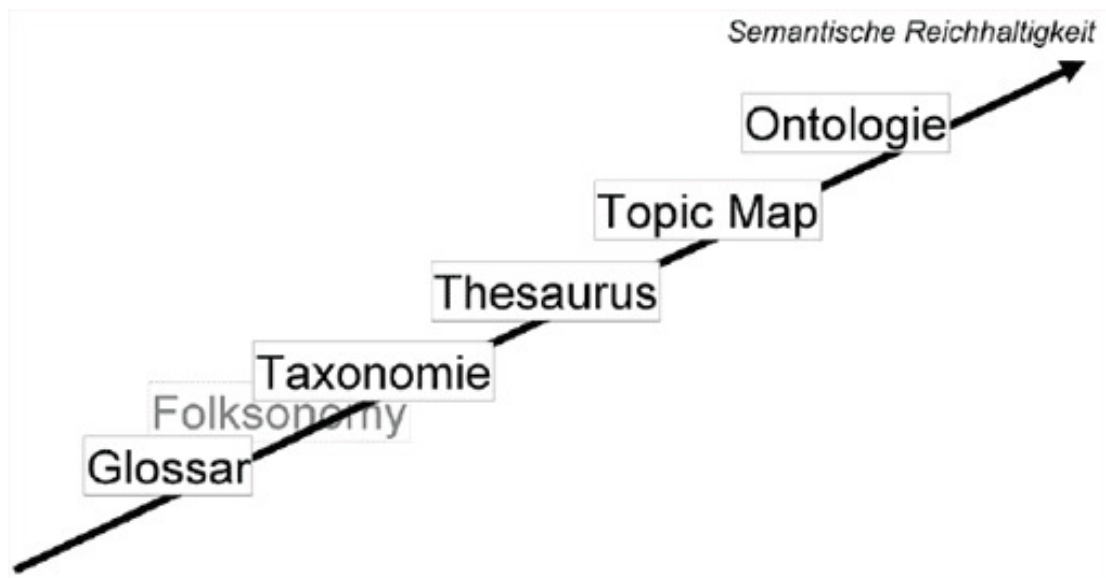


Abb. 3: Semantische Treppe³⁷

Die einfachste Form ist ein Glossar, in dem Begriffe alphabetisch gesammelt und erklärt werden. Etwas schwieriger einzuordnen, ist die Folksonomy, da sie sich sehr vielfältig präsentiert. Auf der Abbildung ist zu erkennen, dass die Ontologien wesentlich komplexer als Taxonomien und Thesauren sind. Jedoch ist der Begriff der Ontologie ein sehr weitläufiger, da es mittlerweile eine Menge an Modellierungen gibt, die als Ontologie bezeichnet werden. Es existieren die folgenden Unterscheidungen:

³⁵ Vgl. Blumauer/ Pellegrini (2006), S. 10ff.

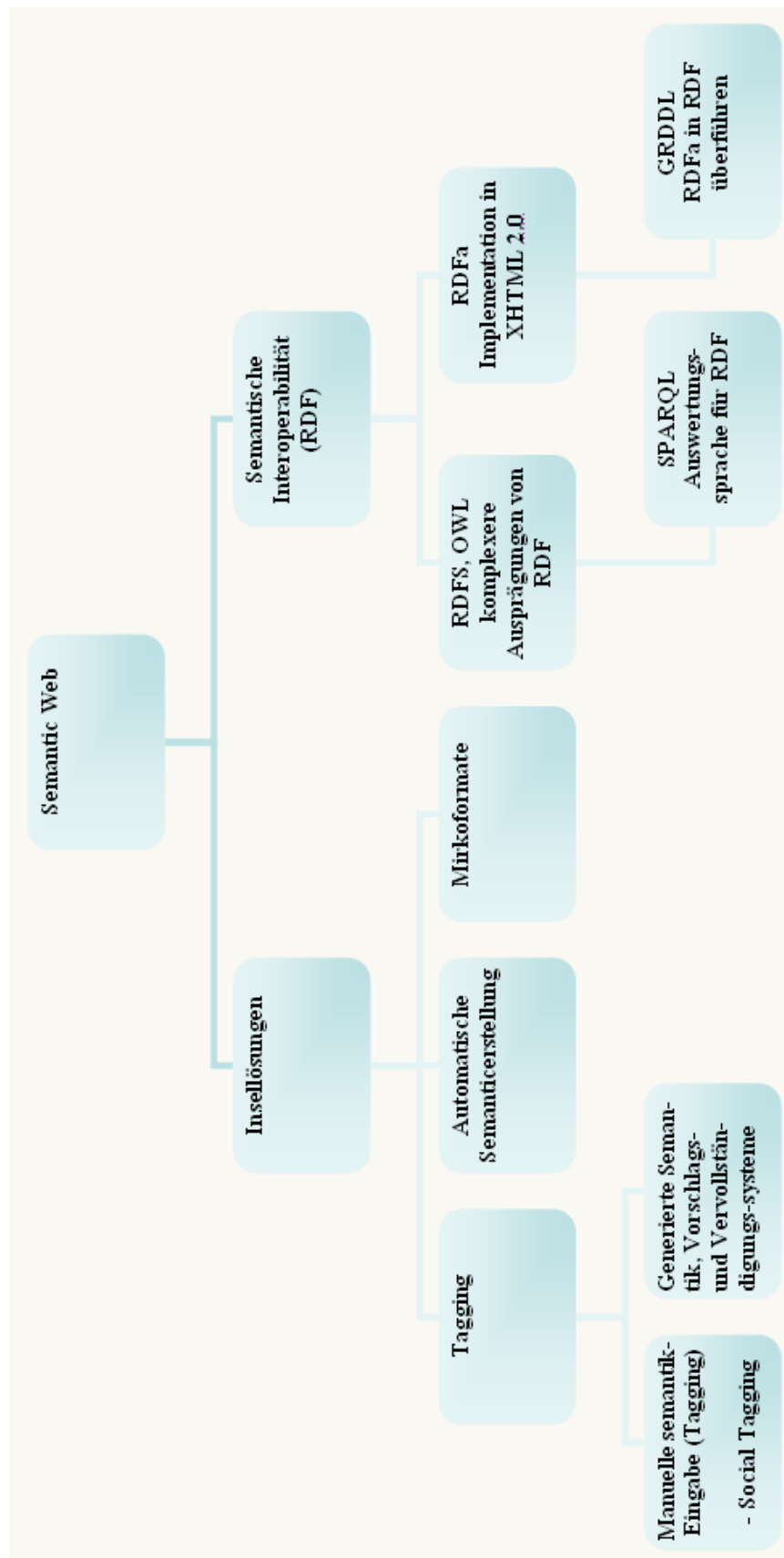
³⁶ Vgl. Gotter (2006).

³⁷ Quelle: <http://wissensexploration.de/img/semantische-treppe.gif>.

- Domänenontologien versuchen das Wissen einer einzelnen Domäne zu erfassen.
- Metadaten-Ontologien sind hingegen Vokabulare, welche zur Beschreibung von Informationsquellen genutzt werden.
- Upper-Level-Ontologien sind generisch und versuchen das Allgemeinwissen zu beschreiben. Sie dienen daher als Basis für einzelne Domänen.
- Light-Weight-Ontologien versuchen durch verschiedene Vokabulare wie SKOS Thesauren zu beschreiben. Vokabulare werden im nächsten Kapitel beschrieben

3.2 Entwicklungsformen / Ausprägungen

Ein prinzipieller Bedarf für das semantische Web ist folglich vorhanden, doch wie sieht die Realität aus? Im Wesentlichen gibt es zwei Entwicklungsrichtungen: Die eine möchte ein Web 3.0 einführen, was bedeutet, dass alle Webseiten semantische Schnittstellen besitzen. Diese ganzheitliche Lösung wird vor allem vom World Wide Web Consortium seit Jahren vorangetrieben. Die andere Entwicklung stellt weniger den prinzipiellen Nutzen als den konkreten Bedarf in den Vordergrund und es entstehen kleine Insellösungen, welche sich aber als nützliche Helfer erweisen.

Abb. 4: Struktur und Zusammenhang der theoretischen Modelle³⁸³⁸ Quelle: Eigene Darstellung.

3.2.1 Insellösungen

Die Entwicklung des so genannten Web 3.0 gestaltet sich aufwändig und langsam. Da die Insellösungen sich am konkreten Bedarf orientieren und punktuell umgesetzt werden, erfreuen sie sich zunehmender Popularität.³⁹ Dabei gibt es verschiedene Ansätze:

3.2.1.1 Tagging

Der Vorgang Tagging, also das Setzen von Tags, beschreibt das Hinzufügen von Schlagworten zu Inhalten von Webseiten, wie zum Beispiel Newsartikeln, Bildern oder anderen Objekten. Das Setzen von Tags geschieht in den überwiegenden Fällen von Hand und erfordert dementsprechend Fleißarbeit der Benutzer. Auf vielen Webseiten sind Tags oder auch andere semantische Techniken schon vorhanden, wodurch der Bedarf deutlich wird.⁴⁰ Die Vorteile ergeben sich dadurch, dass der Benutzer Inhalte mit seinen Begriffen einordnen kann und auf diese Weise das Wiederfinden erleichtert wird. Außerdem können Inhalte auf diese Art und Weise miteinander verbunden werden.

Tagging hat heute eine sehr große Verbreitung gefunden und ist auf vielen Seiten wie welt.de⁴¹ oder sueddeutsche.de⁴² zu finden. Diese Internetseiten stellen ihre News mit einem Bereich für Tags aus⁴³, sodass mit deren Hilfe der Eintrag besser wiedergefunden werden kann. Dieser Bereich lässt sich unter dem Terminus Tag Clouds definieren, sogenannte Schlüsselwortwolken. In diesen Wolken werden zum Thema passende, häufig benutzte Tags in meist alphabetischer oder thematischer Reihenfolge aufgelistet. Ein treffendes Beispiel dafür zeigt auch focus.de⁴⁴. Dort werden die wichtigen Tags nicht nur in einem Bereich gesammelt, sondern je nach Häufigkeit in verschiedenen Größen dargestellt. Die Buchstabengröße wird hierbei logarithmisch errechnet. Wenn noch die Möglichkeit besteht einen zeitlichen Faktor einfließen zu lassen, enthält die Tag Cloud einen guten Überblick über die aktuellen Themen und deren Relevanz.⁴⁵ Eine weitere Option besteht darin, die Tags in der Wolke nicht alphabetisch zu ordnen, sondern nach Nähe. Dabei muss ermittelt werden wie eng Begriffe zueinander stehen. Dies geschieht zum Beispiel durch Data Mining.⁴⁶

Social Tagging

Dass die vergleichsweise einfache Vorgehensweise des Taggings sehr effektiv sein kann, zeigen die sozialen Bookmarkdienste Mister Wong⁴⁷ und delicious⁴⁸. Dort kann jeder Anwender eine

³⁹ Vgl. Pellegrini (2008), S. 22ff.

⁴⁰ Vgl. Alby (2007), S 117.

⁴¹ <http://www.welt.de>, letzter Zugriff am 16.07.2009.

⁴² <http://www.sueddeutsche.de/>, letzter Zugriff am 16.07.2009.

⁴³ Vgl. Schwarz (2008).

⁴⁴ <http://www.focus.de/>, letzter Zugriff am 16.07.2009.

⁴⁵ Vgl. Gaiser (2008), S. 17.

⁴⁶ Vgl. ebd., S. 27.

⁴⁷ <http://www.mister-wong.de/>, letzter Zugriff am 16.07.2009.

Webseite im Internet eintragen und diese mit Schlüsselwörtern, die er mit der Webseite assoziiert, versehen. Auf diese Weise ist im Fall von Mister Wong eine Datenbank von über 6 Millionen Lesezeichen entstanden, welche mit Hilfe der Schlüsselwörter einfach wiedergefunden werden können. Die Lesezeichen beziehen sich sowohl auf die Top-Level Adressen der Webseiten als auch auf konkrete Unterseiten, die sich mit einem bestimmten Thema befassen. Durch die Angabe von Tags werden diese Seiten in einen Kontext zueinander gebracht. Während bei herkömmlichen Suchmaschinen die Suchworte in textueller Form auf einer Seite vorhanden sein müssen, ist dies beim Tagging nicht mehr nötig. Der Benutzer kann die Seite mit dem assoziierten Tag wiederfinden. Wird beispielsweise bei delicious nach den Tags „Joomla template free“ gesucht, zeigt die Webseite eine Übersicht mit Links, welche mit diesen Schlagwörtern annotiert wurden. Auch ist ersichtlich, wie viele User sich diese Seite als Lesezeichen gespeichert haben, wodurch die Ergebnisse eine bestimmte Gewichtung bekommen.

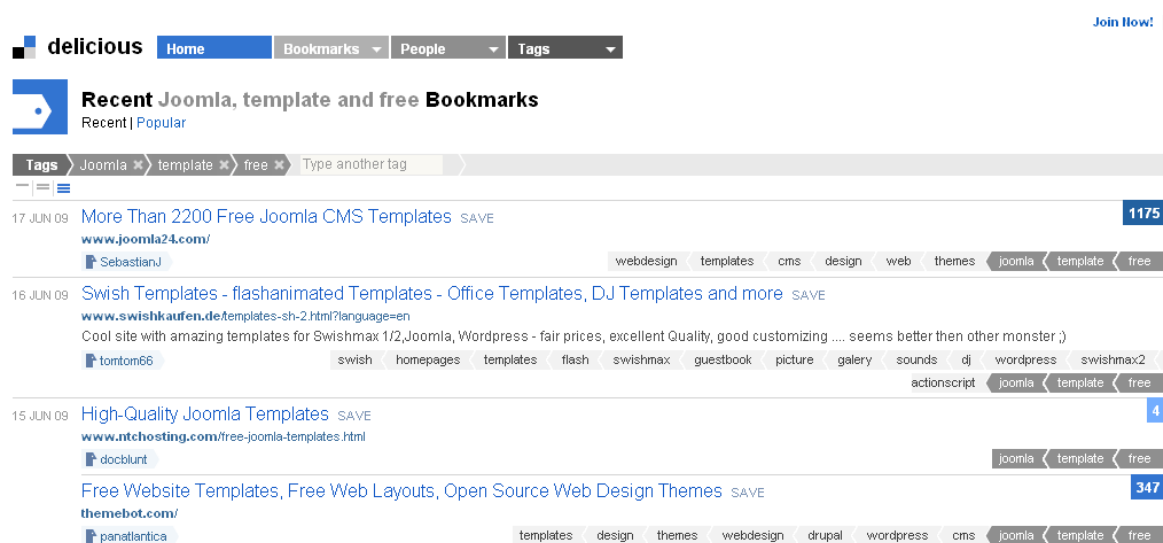


Abb. 5: Suchergebnis von delicious⁴⁹

Im Gegensatz dazu sind die Ergebnisse bei der Suchmaschine Google⁵⁰ nicht immer aussagekräftig, da hier auch Ergebnisse wie Mister Wong angezeigt werden, welche nur einen indirekten Bezug zu diesen Tags haben.

Social Tagging hat den Vorteil, dass die Inhalte einer Webseite präziser und nutzerorientierter auffindbar sind, als mit herkömmlichen Suchmaschinen, welche nur begrenzt automatisiert die Bedeutung einer Webseite erkennen. Diesen Vorteil machten sich unter anderem die Webseiten YouTube.de (Online Video-Portal)⁵¹ und flickr (Online Bilddienst)⁵² zu Nutze, in denen der Anwender nach dem Anlegen eines Objektes Tags vergeben kann. Durch die große Menge an Inhalten ist für

⁴⁸ <http://delicious.com/>, letzter Zugriff am 16.07.2009.

⁴⁹ Quelle: <http://delicious.com/tag/Joomla%20template%20free?detail=3>, letzter Zugriff am 16.07.2009.

⁵⁰ <http://www.google.com/search?client=opera&rls=de&q=Joomla+template+free&sourceid=opera&ie=utf-8&oe=utf-8>, letzter Zugriff am 16.07.2009.

⁵¹ <http://www.youtube.de>, letzter Zugriff am 16.07.2009.

⁵² <http://www.flickr.de>, letzter Zugriff am 16.07.2009.

den Benutzer eine fest vorgegebene Kategorisierung nicht zweckmäßig. Im Zuge des Web 2.0, in dem jeder Benutzer am Internet mitwirken kann, konnten diese Systeme große Erfolge verbuchen beziehungsweise Schwierigkeiten, die solche Projekte aufweisen, durch die große Masse an Benutzern reduzieren.

Dennoch ist auch Social Tagging nicht frei von Problemen. Einige Portale, wie Mister Wong, ermöglichen es nicht mehrere Wörter als einen Begriff zu taggen, da sie die Tags anhand von Leerzeichen trennen. Der Tag „Semantik Web“ würde also zu zwei Tags aufgespalten. Daraus resultiert, dass Benutzer dazu übergehen das Leerzeichen durch andere Zeichen zu ersetzen. „Semantik_Web“, „Semantik-Web“, „SemantikWeb“ sind mögliche Beispiele. Das Problem besteht darin, dass dies drei verschiedene Tags sind, die aus Systemsicht nichts miteinander zu tun haben. Ein ähnliches Problem tritt auf, wenn sich Wörter auf Grund von Rechtschreibfehlern unterscheiden. Ein sehr bekannter und im Internet weit verbreiteter Fehler ist das Wort „Standard“, welches gerne als „Standart“ geschrieben wird. Auch hier können Bookmarks, Videos oder Bilder, die mit „Standart“ getaggt wurden nicht unter dem richtigen Begriff „Standard“ gefunden werden.

⁵³ Neben den unterschiedlichen Schreibweisen von Tags gibt es dazu noch generellere Probleme. Jeder Benutzer taggt Inhalte mit den Schlagworten, die er in dem Moment damit assoziiert. Weil jedoch jeder Mensch auf Grund seiner Erfahrungen eine andere Sichtweise auf ein Thema hat, können sich die Tags von Autor zu Autor unterscheiden. Gleiches gilt wenn ein Autor im Laufe der Zeit seinen Blickwinkel auf ein Thema ändert, da er neues Wissen über dieses Thema erlangt hat und nach einer gewissen Zeit andere, spezifischere Tags vergeben würde. In der menschlichen Gehirnstruktur ändern sich solche Verknüpfungen zu Themen automatisch, bei Tagging Systemen muss dies manuell nachgepflegt werden. Tags sind also immer zeit-, themen- und situationsgebunden. Daher werden Tags folgendermaßen definiert: Sie bestehen aus einem Objekt (dem Link), einem Label (der Beschriftung) und dem User der den Tag vergeben hat. Zusätzlich können auch noch weitere Daten wie das Datum erfasst werden. Eine Möglichkeit diese Probleme anzugehen liegt darin Wörter vorzuschlagen.⁵⁴

Doch wie konnten sich trotz dieser Hürden Systeme wie Mister Wong oder flickr mit ihrem Fokus auf Tagging durchsetzen? Der Schlüssel liegt in den so genannten Folksonomies. Das Wort Folksonomy setzt sich aus den englischen Wörtern „Folk“ und „Taxonomy“ zusammen und bezeichnet die gesamte Stimme aller Benutzer.⁵⁵ Auch wenn es in diesen Systemen keinen Experten gibt, der eine Ordnung erstellt, entwickelt sich diese durch Selbstorganisation. Über die Zeit und die Masse kristallisiert sich eine einheitliche Bedeutung und Verwendung von Begriffen heraus. Da sich die Tags eines Objektes einem Benutzer zuordnen lassen, konnte beobachtet werden, dass Benutzer ähnlicher Interessen mit der Zeit ähnliche Begriffe zur Tagvergabe benutzten.⁵⁶ Aber auch die Folksonomies können unter dem Problem leiden, dass in der Regel davon auszugehen ist, dass 90%

⁵³ Vgl. Klein (2006).

⁵⁴ Vgl. Gaiser (2008), S. 26F, 45ff, 141ff.

⁵⁵ Vgl. ebd., S.20.

⁵⁶ Vgl. ebd., S.180.

der Benutzer eines Systems die Inhalte nur verwenden. 9% erstellen gelegentlich Inhalte und 1% sind für den größten Teil der Inhalte des Web 2.0 verantwortlich.⁵⁷

Vorschlags- und automatische Vervollständigungssysteme

Daher liegt es nahe einen objektiven und automatischen Mechanismus zu entwickeln, um Tags aus bestehenden Dokumenten abzuleiten. Ein einfaches Beispiel hierfür ist die Seite webnews.de⁵⁸, wenn dort von einer Person ein neuer Eintrag verfasst wurde, soll dieser Benutzer den Beitrag mit Tags versehen. Dazu werden ihm häufig benutzte Tags aus dieser Kategorie vorgeschlagen, die er per Klick hinzufügen kann. Dies führt zu einer einheitlicheren Tagvergabe.

Den nächsten Schritt in diese Richtung weist die Seite [calais](http://calais.com)⁵⁹. Mit Hilfe der angebotenen Online-Applikation ist es möglich Texte zu analysieren und Tags zu generieren. Die Applikation erkennt zurzeit englische Begriffe und Texte und ordnet sie den 3 Oberkategorien „bekannte Objekte“ wie Menschen, Firmen, Orte; „Fakten“, wie Ausbildungen oder Politik und „Ereignisse“, wie Sport oder Forschung zu. Ebenso erkennt das Programm Personen, Beziehungen zwischen Personen und Zitate. Da sich diese Applikation in verschiedene Systeme wie Blogs, CMS und anderen Webapplikationen einbauen lässt (Wordpress⁶⁰, Drupal⁶¹) wird so die Möglichkeit geboten einen Text automatisch mit Semantik zu versehen. Dabei hat der Anwender die Möglichkeit zu entscheiden, ob die vorgeschlagenen Tags sofort übernommen werden, oder ob sie dem Benutzer als Tagvorschläge präsentiert werden.

Ein ähnliches Prinzip verfolgt auch die Seite [Zemanta](http://zemanta.com)⁶². Hier wird ebenfalls der, durch die API oder durch das Einbinden einer Komponente in ein System wie Wordpress, übergebene Text analysiert. Der Algorithmus erkennt Schlagwörter wie Namen oder Orte und sucht automatisch dazu passende Bilder bei flickr, Einträge bei Wikipedia, Webseiten zu erwähnten Objekten im Text und natürlich auch Tags heraus. Auch wenn die Tags und Einträge auf Wikipedia auf Englisch sind, funktioniert die Analyse ebenfalls mit anderssprachigen, wie zum Beispiel deutschsprachigen Texten.

3.2.1.2 Automatische Semanticerstellung

Auf eine etwas andere Art geschieht die semantische Nutzung bei einigen Social Networks wie zum Beispiel facebook⁶³. Nachdem der Benutzer seine Daten, zumeist freiwillig und großzügig, in

⁵⁷ Vgl. Institut für Software-Ergonomie und Usability (2006).

⁵⁸ <http://www.webnews.de>, letzter Zugriff am 16.07.2009.

⁵⁹ <http://www.opencalais.com/>, letzter Zugriff am 16.07.2009.

⁶⁰ <http://wordpress.org/>, letzter Zugriff am 16.07.2009.

⁶¹ <http://drupal.org/>, letzter Zugriff am 16.07.2009.

⁶² <http://www.zemanta.com/>, letzter Zugriff am 16.07.2009.

⁶³ Vgl. Adewumi (2008).

das System eingegeben hat, wertet facebook⁶⁴ diese aus und schlägt Personen vor, die in Verbindung zum eigenen Profil stehen könnten. Dabei scheint facebook Daten wie Ort, Schule und Freunde von anderen Freunden auszuwerten und in Zusammenhang zu setzen. Es hat sich gezeigt, dass das Prinzip sehr gut funktioniert und der Benutzer insbesondere nach der Neuregistrierung mit geringem Aufwand seinen Freundeskreis finden kann.

Ein ähnliches Verhalten ist auch bei der Social Network Seite badoo⁶⁵ zu finden, welche anhand des Benutzerverhaltens versucht eine Verbindung herzustellen. Dazu wird ausgewertet, ob ein Benutzer ein Profil betrachtet hat, ein Bild des Profils bewertete oder in der Nähe wohnt. Diese Aktionen werden als mögliche Indikatoren genutzt um herauszufinden, ob es einen Zusammenhang zwischen diesen Personen geben könnte. Auch wenn der Zusammenhang durch eine Bewegungsanalyse häufig noch zu Fehlinterpretationen führt, ist es eine benutzerfreundliche Möglichkeit. Dennoch erzielt eine Auswertung von Tags, wie sie bei dem Social Network Habbo⁶⁶ für alle Profile, Gruppen und anderen Elementen durchgeführt wird, zurzeit eine größere Trefferquote. Einen Mittelweg aus diesen genannten Elementen, versuchen andere Websites für sich zu erschließen. Die Seite Flixster⁶⁷ ist ein Social Network für alle Filminteressierte, hier wird der Anwender aufgefordert seine Lieblingsfilme einzugeben und zu bewerten. Anhand dieser Filme werden ähnliche Filme herausgesucht, welche der Benutzer kennen könnte und dann entsprechend bewerten soll. Dadurch, dass die Bewertung sehr einfach ist und der Benutzer durch Filme, welche er gut kennt, motiviert wird weitere Bewertungen abzugeben, lernt das System den Benutzer Schritt für Schritt besser kennen. Die vorgeschlagenen Filme waren sehr häufig passend und reichten von sehr aktuellen Filmen bis hin zu Klassikern.

Ein ähnliches Prinzip verfolgt auch die Seite StumbleUpon⁶⁸, deren Ziel ist es die besten Seiten des Internets zu kennen. Dazu gibt der Benutzer nach der Registrierung seine Interessen ein und installiert ein kleines Tool für den Browser. Mit Hilfe dieses Tools werden dem Benutzer, anhand der gewählten Kategorien, auf Knopfdruck Seiten gezeigt, die für ihn interessant sein können. Diese Seiten kann jeder Benutzer vorher anlegen und mit Tags versehen. Nach Aufruf der Seite kann der Benutzer, ähnlich wie mit der Google Toolbar⁶⁹, die Seiten bewerten, wodurch eine persönliche Tag Cloud begehrter Themen entsteht.

Nach dem Prinzip des Bewertens funktioniert auch die Seite webnews.de. Hier können Nachrichtenberichte beurteilt werden. Außerdem stehen neben jedem Nachrichteneintrag Beiträge mit ähnlichen Themen und Beiträge, welche andere Benutzer gut fanden, die diesen Artikel gelesen haben. Zusätzlich werden die Bewertungen analysiert und daraus dem Benutzer Profile von Leuten präsentiert, die ähnliche Interessen haben. Diese Seite ist ein gutes Beispiel dafür, dass semantische Zusammenhänge aus den Aktionen eines Benutzers interpretiert werden können.

⁶⁴ <http://www.facebook.com/>, letzter Zugriff am 16.07.2009.

⁶⁵ <http://badoo.com/>, letzter Zugriff am 16.07.2009.

⁶⁶ <http://www.habbo.de/>, letzter Zugriff am 16.07.2009.

⁶⁷ <http://www.flixster.com/>, letzter Zugriff am 16.07.2009.

⁶⁸ <http://www.stumbleupon.com/>, letzter Zugriff am 16.07.2009.

⁶⁹ <http://toolbar.google.com/>, letzter Zugriff am 16.07.2009.

Professioneller und auf den kommerziellen Nutzen basierend, wird Semantik bei amazon.de⁷⁰ eingesetzt. Auch dort sind alle Artikel mit Tags versehen, nach denen der Benutzer suchen kann. Die Arbeit des Taggens wird jedoch größtenteils von dem Unternehmen übernommen. Zusätzlich zum Tagging werden hier im großen Stil Benutzeraktionen ausgewertet. Dadurch werden dem Benutzer bei Ansicht eines Artikels weitere Artikel vorgeschlagen, welches in mehrfacher Form geschieht: Zum einen, welche Artikel in Verbindung mit diesem Produkt gekauft wurden, außerdem welche Artikel nach Ansicht dieses Produktes bestellt wurden und welche Erzeugnisse bevorzugt zusammen erworben werden. Diese drei Kategorien haben in der Regel eine gute Korrelation zum gerade angesehenen Produkt. Der Vorteil dieser Auswertung ist, dass der Benutzer keine expliziten Aktionen wie das Anklicken einer Bewertung machen muss, sondern dies aus dem normalen Verhalten herausgefiltert werden kann. Außerdem schlägt amazon.de auch Produkte vor, die zu dem passen, was die Person schon einmal dort erworben hat. Allerdings fehlt hier eine zeitliche Komponente, so dass auch Artikel empfohlen werden, welche vor Jahren gekauft wurden.

3.2.1.3 Ausblick

Zusammenfassend wurde festgestellt, dass die häufigste Anwendung von Semantik im Web darauf abzielt ähnliche Objekte zu finden und zu verknüpfen. Dazu gesellen sich zu den bisher erwähnten Quellen eine Menge anderer bekannter Seiten. Falls eine Person einen Gegenstand bei ebay⁷¹ nicht ersteigern konnte, werden dem Nutzer ähnliche Auktionen empfohlen. Wenn eine Person bei dem Musikportal last.fm⁷² eine Musiksammlung mit seinen Lieblingskünstlern anlegt, bekommt der Nutzer Empfehlungen zu Künstlern ähnlicher Musikrichtung. Demzufolge lassen sich noch weitere Seiten aufzählen, welche dieses Prinzip verwenden. Aus diesen vielen Beispielen lässt sich schlussfolgern, dass das Semantic Web schon jetzt eine große Verbreitung in Form von Tags gefunden hat. Dazu kommt die Auswertung der schon vorhandenen Daten wie Adressdaten. Zu diesen Daten werden ebenfalls Verbindungen hergestellt und, wenn möglich, von der Werbewirtschaft genutzt. Denn gerade für diese ist es relevant ihre Produkte Zielgruppen genau zu platzieren. Als Beispiel sei hier der Datenschutzstreit⁷³ bei dem Social Network studiVZ⁷⁴ erwähnt obwohl auch andere Unternehmen⁷⁵ wie GMX⁷⁶ Daten zu Geschäftszwecken auswerten.

Die Semantik auf einer Webapplikation automatisch generieren zu lassen ist dagegen noch nicht weit verbreitet. Erste Ansätze sind bei amazon.de zu finden, wo die Bewegungsprofile ausgewertet werden. Ebenfalls ein guter Ansatz scheinen die Projekte Calais oder Zamanta zu sein. Dennoch hat sich das Vergeben von Tags, insbesondere in den so genannten Web 2.0 Anwendungen, gut etabliert und ist von den Benutzern akzeptiert worden. Von Benutzern erstellte Inhalte können

⁷⁰ <http://www.amazon.de/>, letzter Zugriff am 16.07.2009.

⁷¹ <http://www.ebay.de/>, letzter Zugriff am 16.07.2009.

⁷² <http://www.lastfm.de/>, letzter Zugriff am 16.07.2009.

⁷³ Vgl. Lischka (2007).

⁷⁴ <http://www.studivz.net/>, letzter Zugriff am 16.07.2009.

⁷⁵ Vgl. GMX (2009).

⁷⁶ <http://www.gmx.net/>, letzter Zugriff am 16.07.2009.

leicht durch Tags semantisch erfasst werden, wenn eine große Community etabliert ist. Ein kollaboratives Lernportal wie learnuzz ist demnach für Tags prädestiniert. Allerdings muss bei einer Semantik auf Tag-Basis auf die Selbstorganisation der Folkonomien vertraut werden. Diese muss nicht immer optimal sein und es kann auch nicht unterschieden werden, ob ein Tag „BWL“ einen Download oder eine Gruppe meint. Für das System sind die Tags nur Zeichenfolgen die mit Benutzern und Objekten in Verbindung gebracht werden. Statt der Selbstorganisation gibt es aber auch die Möglichkeit des ontologiebasierten Taggens. Dem Thema Ontologien widmet sich das Kapitel der W3C Standards, denn zuvor sollte geklärt werden was Semantic Web genau umfasst.

3.2.2 Semantic Web

„*The Semantic Web is a web of data.*“ (<http://www.w3.org/2001/sw/>, letzter Zugriff am 16.07.2009). Damit ist nicht gemeint, dass es ein zweites World Wide Web geben soll. Das Ziel ist es das bestehende Web mit Semantik anzureichern. Diese Idee kam schon Ende der neunziger Jahre auf, also noch vor dem großen Internet-Boom am Anfang des neuen Jahrtausends. Um dies zu erreichen gibt es verschiedene Ansätze. Zum einen ist die Organisation microformats⁷⁷ entstanden, welche sich zum Ziel gesetzt hat, auf der jetzigen Technik aufzubauen und schnelle, einfache Konzepte zur Integration von Semantik zu erstellen.⁷⁸ Auf der anderen Seite beschäftigt sich das W3C schon seit Jahren damit eine umfassende semantische Lösung zu definieren. Die semantische Lösung besteht aus mehreren Komponenten, welche zum Teil erst seit kurzem als offizielle Empfehlungen des W3C gelten.

3.2.2.1 Microformats

Es wurde bereits erklärt, dass sich Daten im Internet sehr gut und einfach durch Tags beschreiben lassen. Dadurch wird ein Dokument oder Objekt genauer beschrieben. Aber auch wenn die Idee mit dem Tagging aufgegriffen wird, führt dies zwar dazu, dass die Maschine ein Dokument inhaltlich einordnen kann, das Dokument aber immer noch nicht verstehen kann. Durch Tags könnte ein Termin derartig gekennzeichnet werden, dass er zum Beispiel im März stattfindet, dass es sich um eine Musikgruppe handelt. Aber es ist für die Maschine trotzdem ein Datenobjekt, kein Termin, der weiterverarbeitet werden kann und sich automatisch in den Kalender importiert. Diesem Ziel widmet sich zum einen das W3C, in dem es das große Semantische Web entwickelt, mit einem theoretischen Hintergrund und allgemeinen Konzepten welche trotzdem speziell genug sind, als dass sie für alle Situationen angewandt werden können. Doch es gibt auch andere Wege.

⁷⁷ <http://microformats.org/>, letzter Zugriff am 16.07.2009.

Was sind microformats

Dieses große Ziel eines allgemeinen Standards hat microformats (kurz μF) nicht, weshalb es sich auch als das semantic web, also in Kleinschreibung, bezeichnet.⁷⁹ Für ein bestimmtes Problem, zum Beispiel Termine, werden konkrete Abbildungsvorschriften entwickelt, mit welchen die meisten Anwendungsfälle gelöst werden können. Die Lösung einfacher Probleme steht im Vordergrund und die Technik soll auf den aktuellen Standards aufbauen.⁸⁰ Dies wird getreu dem Pareto Prinzip, mit geringem Aufwand viele Probleme lösen, umgesetzt. Microformats werden in erster Linie für den Menschen entwickelt, in zweiter Linie für die automatische Auswertung.⁸¹ Denn auch wenn Informationen semantisch annotiert werden, wird durch das Auszeichnen nichts versteckt, alle Daten bleiben weiterhin für den Menschen lesbar. Wichtig ist dabei: „*DRY: Don't repeat yourself*“ (<http://microformats.org/wiki/dry>, letzter Zugriff am 16.07.2009). Es wäre viel einfacher, wenn ein Mensch einen Termin nur einmal auf seiner Website eintragen muss und andere Systeme ihn dann auslesen, als dass er diesen Termin bei verschiedenen Portalen manuell eintragen muss wie es zur Zeit der Fall ist. Aufgrund ihrer Einfachheit bergen microformats allerdings auch ein paar Nachteile. So wird kein gemeinsames Datenmodell für die verschiedenen Vokabulare entwickelt sondern jedes Mal eine neue Abbildungsvorschrift. Auch ist die Vokabularstruktur sehr flach, da es keine Namensräume gibt.⁸² Warum setzen dann aber sowohl kleine Webseiten wie last.fm als auch große Firmen wie Yahoo mit ihrem Projekt Search Monkey⁸³ auf microformats und bekennen sich zum „Semantic Web light“?⁸⁴

Anwendungsmöglichkeiten

Der Grund dafür ist verständlich, denn microformats sind einfach aufgebaut. Die Metadaten werden in HTML-Attributen angegeben. Zuerst muss das XHTML Meta Data Profile, ein Name-Wert-Paar Profil, im Head-Bereich der Webseite angegeben werden, um das Profil für die Seite festzulegen.⁸⁵ Es gibt die verschiedensten Arten von microformats: Zum einen die sehr einfach gehaltenen. Dazu zählt zum Beispiel XFN, durch welches ermöglicht wird mittels eines rel-Attributs im a-Tag⁸⁶ einer Website eine Verbindung zu einer anderen Person oder Zielseite zu kennzeichnen.⁸⁷ Es ist auch möglich durch dieses rel-Attribut Abstufungen wie „contact“, „acquaintance“ und „friend“⁸⁸

⁷⁸ Vgl. Microformats (2009), <http://microformats.org/about/>, letzter Zugriff am 16.07.2009.

⁷⁹ Vgl. Maurice (2009), S. 65.

⁸⁰ Vgl. ebd., S. 67.

⁸¹ Vgl. Allsopp (2007), S. 9.

⁸² Vgl. Blumauer/ Pellegrini (2009), S. 151ff.

⁸³ Mit diesem Tool können Suchergebnisse um Informationen erweitert werden, welche mittels microformats, eRDF, RDFa, RSS oder Atom von der Webseite ausgelesen wurden.
<http://developer.yahoo.com/searchmonkey/>, letzter Zugriff am 16.07.2009.

⁸⁴ Vgl. Blumauer/ Pellegrini (2009), S. 7.

⁸⁵ Vgl. Maurice (2009), S. 68ff.

⁸⁶ rel und a sind HTML-Befehle, vgl. Münz/ Nefzger (1999), S. 75f, S. 400.

⁸⁷ Vgl. Maurice (2009), S. 71.

⁸⁸ „contact“ ist englisch für Kontakt, „acquaintance“ steht für Bekanntschaft und „friend“ für Freund.

darzustellen. Eine Schwäche ist aber auch hier schnell erkennbar, da für das Ziel des Links nicht zwischen einer Person und einer Website unterschieden wird.⁸⁹ Für dieses microformat ist also nichts weiter nötig als im Header der Seite das Profil anzugeben und den Link um das entsprechende rel-Attribut zu erweitern. Auch Schlagwörter, können, wenn sie als Link dargestellt werden, als Tag gekennzeichnet werden, in dem der Link um das Attribut rel="tag" erweitert wird.

Ebenso einfach lassen sich auch Vote-Links⁹⁰ kennzeichnen und zwar mit dem selten benutzen Attribut „rev“ als Rückwärtslink. In diesem Fall ist dieses wenig verwendete Attribut notwendig, denn es ist ein Link zu einer Bewertung über die aktuelle Seite und nicht eine Beschreibung über die Seite des Links. Auch hier gibt es mit „vote-for“, „vote-against“⁹¹ und weiteren Attributwerten die Möglichkeit den Link konkret zu beschreiben.⁹²

Neben noch weiteren einfachen Links wie für Lizenzen⁹³ oder Verzeichnisauflistungen gibt es auch die zusammengesetzten microformats, welche den eigentlichen Schwerpunkt bilden. Diese Art von microformats werden durch die Verschachtelung verschiedener Klassen, wie es von CSS⁹⁴ bekannt ist, dargestellt. Es gibt eine Oberklasse zur Einleitung sowie verschiedene Unterklassen. Damit es nicht zur Kollision in Tags kommt, die schon aus Designgründen mit Klassennamen versehen werden, können einem Tag auch mehrere Klassen zugewiesen werden.

Zu den gebräuchlichsten Formaten gehört der hCalender⁹⁵, mit dem Termine, Ereignisse und Veranstaltungen ausgezeichnet werden können. Dieses Format basiert auf dem Standard von iCalendar. Es nutzt das ISO 8601-Format für Datumsangaben (Jahr, Monat, Tag) und wird in Verbindung mit hResume, für Lebensläufe, adr für Adressen und hCard für Profile angegeben. Die hCard⁹⁶ leitet sich von der vCard vom Internet Mail Consortium (IMC) ab und ist das zweite häufig genutzte Format. Es dient zur Kennzeichnung von Personen, Organisationen und zur Beschreibung von Lokalisationen. Die Kennzeichnung von Personen lässt sich mit XFN, also der Kennzeichnung von Freunden, kombinieren. Hier ein Beispiel für hCard:

```
„<div class="vcard">  
  
  <a class="fn org url" href="http://www.commerce.net/">CommerceNet</a>  
  
  <div class="adr">  
  
    <span class="type">Work</span>:  
  
    <div class="street-address">169 University Avenue</div>  
  
    <span class="locality">Palo Alto</span>,  
  </div>
```

⁸⁹ Vgl. Blumauer/ Pellegrini (2009), S 221.

⁹⁰ Vote-Links sind links, mit denen abgestimmt werden kann.

⁹¹ „vote-for“ bedeutet dafür stimmen, „vote-against“ dagegen stimmen.

⁹² Vgl. Maurice (2009), S. 93ff.

⁹³ Vgl. ebd., S. 95.

⁹⁴ Vgl. Münz/ Nefzger (1999), S. 101.

⁹⁵ Vgl. Maurice (2009), S. 131ff.

```
<abbr class="region" title="California">CA</abbr>
<span class="postal-code">94301</span>
<div class="country-name">USA</div>
</div>
<div class="tel">
  <span class="type">Work</span> +1-650-289-4040
</div>
<div class="tel">
  <span class="type">Fax</span> +1-650-289-4041
</div>
<div>Email:
  <span class="email">info@commerce.net</span>
</div>
</div>
```

(Microformats (2009): <http://microformats.org/wiki/hcard>, letzter Zugriff am 16.07.2009)

Neben den neun fertigen Spezifikationen hCalendar, hCard, rel-license, rel-nofollow, rel-tag, VoteLinks, XFN, XMDP und XOXO⁹⁷ gibt es noch eine ganze Reihe weiterer in Entwicklung befindender Formate.⁹⁸ Mit adr lassen sich einfache Adressen, bei denen der Name oder die Organisation unbekannt sind, auszeichnen. Adr ist ebenso wie geo⁹⁹ eine Ausgliederung von hCard, mit dem sich Orte mittels Längen- und Breitengrad kennzeichnen lassen. Es gibt noch viele weitere wie hAudio für Audioformate, hAtom für syndizierte Inhalte wie Blogs oder News, oder xFolk für Bookmarklisten. Jeder hat die Möglichkeit neue Vorschläge zu machen, jedoch schaffen es nur wenige Formate aufgenommen zu werden, da sie sich dazu in der Praxis bereits bewährt haben müssen.

Ein etwas anders geartetes Format ist WebSlices¹⁰⁰, denn es stammt nicht von der microformats Organisation, sondern von Microsoft. Es arbeitet ähnlich wie hAtom, jedoch wurde es um einige Aspekte ergänzt. Es kann bei schnell ändernden Dingen wie Auktionen, Wetter oder News eingesetzt werden um den aktuellen Status zu überwachen. Da Auktionen enden, können auch WebSli-

⁹⁶ Vgl. Maurice (2009), S. 113ff.

⁹⁷ Vgl. Microformats (2009), http://microformats.org/wiki/Main_Page, letzter Zugriff am 16.07.2009.

⁹⁸ Vgl. Maurice (2009), S. 111f.

⁹⁹ Vgl. ebd., S. 105ff.

¹⁰⁰ Vgl. ebd., S. 35ff, S. 16ff.

ces auslaufen und ungültig werden.¹⁰¹ Allerdings ist dieses Beispiel von Microsoft eher die Ausnahme, da die Firma im Wesentlichen auf die W3C Standards setzt.

Zusammenfassung

Es gibt einige Gründe, die für den Einsatz von microformats sprechen, wie die einfache Integrierbarkeit und Einstiegsmöglichkeit in das semantic web. Allerdings gibt es auch Argumente die dagegen sprechen, wie der eingeschränkte Einsatzbereich und die geringe Standardisierung und die daraus resultierenden Beschränkungen. Einen ausführlichen Vergleich zwischen microformats und dem Äquivalent RDFa des W3C ist auf der Seite von Evan Prodromou zu finden.¹⁰² Welches von beiden Formaten sich durchsetzen wird oder ob sie parallel existieren können ist schwer zu beantworten.

3.2.2.2 W3C Standards

Seit der Definition der allgemeinen Sprache Extensible Markup Language (XML) des W3C bauen viele neue Techniken darauf auf.

XML

Als Grundlage für alle von der W3C definierten Standards für das Semantic Web gilt XML. Dies ist eine Markup Sprache mit der Dokumente ausgezeichnet werden um Textteile zu annotieren. Ähnlich wie HTML baut XML auf Tags auf. Tags sind in diesem Zusammenhang Befehle in der Form „<html>“. In XML sind diese Befehle jedoch nicht zur Darstellung des Dokumentes, sondern zur logischen Strukturierung vorhanden und sehr allgemein gehalten. Daher dient XML auch als ein Austauschformat für Daten, insofern sie serialisiert wurden.

Jede XML Datei besteht aus Elementen der Form „<StartTag>Information</EndTag>“, wobei jeder Tag mit Attributen versehen werden kann <Tag version=“1.0“ /> und die Tags ineinander verschachtelt werden können. Die Struktur einer XML Datei und die verwendbaren Tags werden in der DTD oder in der XML Schema definiert. Dort kann festgelegt werden welche Befehle wie verschachtelt werden können, welche Attribute erlaubt sind und vieles mehr. Die wesentlichen Unterschiede zwischen den beiden Formaten bestehen darin, dass XML Schema auf der XML Syntax beruht und Vererbung sowie komplexe Datentypen erlaubt. Damit ist XML Schema die mächtigere Sprache und der DTD vorzuziehen. Trotz der umfangreichen Möglichkeiten die XML bietet reichen diese nicht aus um Semantik in einem Dokument vollständig darzustellen. Zwar erlaubt XML Markup-Sprachen wie XHTML zu definieren oder Dokumente mittels XSLT in andere Formate zu transformieren, allerdings werden dort immer nur verschachtelte Elemente und somit Bäume und Attribut-Wert-Paare dargestellt. Auch können die einzelnen Tags zwar anhand der XML Scheme

¹⁰¹ Vgl. Blumauer/ Pellegrini (2009), S. 7.

Definition verarbeitet werden, jedoch nicht verstanden oder in Bezug gebracht werden. Für ein Programm, wie zum Beispiel einem Browser, ist es egal ob eine Überschrift <h1> heißt oder <a8z>, solange sie korrekt definiert wurde. Genau dies ist sowohl die Stärke als auch die Schwäche von XML.¹⁰³

RDF

Das Resource Description Format (RDF) ist ein grundlegendes Format des Semantic Web, welches auf XML aufbaut und somit ein XML-Vokabular ist. Die erste offizielle Spezifikation des W3C stammt aus dem Jahr 1999. Die letzte Version, welche auch als Empfehlung des W3C gilt¹⁰⁴, kommt aus dem Jahr 2004. Damit ist RDF einer der älteren Grundsteine der Semantik, findet aber insbesondere durch den momentanen Aufschwung des Semantic Web besondere Bedeutung. In der formalen Sprache wird Wissen durch gerichtete Graphen dargestellt, in denen viele Knoten miteinander verbunden sind. So entsteht ein Netz, welches die Verbindungen und Zusammenhänge darstellt. Als Knoten könnte zum Beispiel das Tier „Jaguar“ existieren. Dieser könnte zum Knoten „Säugetiere“ gehören, welcher mit dem Knoten „Lebewesen“ verbunden ist. In diesem Fall könnte die bevorzugte Struktur eine hierarchische Ordnung sein. Ein Graph würde hingegen mehrfache Verbindungen erlauben, denn der „Jaguar“ ist beispielsweise zusätzlich das Lieblingstier eines kleinen Mädchens namens „Alice“, als auch das Auto ihres Onkels „Bob“. Somit gibt es folgende Verbindungen zu „Jaguar“:

Subjekt	Prädikat	Objekt
Jaguar	gehört zu	Säugetiere
Jaguar	ist Lieblingstier von	Alice
Jaguar	ist Auto von	Bob

Tabelle 1: Einfacher RDF-Graph¹⁰⁵

Die Schwierigkeit besteht nun darin diese Informationen serialisiert darzustellen. Um dies zu leisten, werden alle Kanten des Graphen in der Form „Ausgangsknoten, Kante, Endknoten“ abgespeichert. Diese so genannten Tripel werden in RDF als „Subjekt-Prädikat-Objekt“ bezeichnet.

RDF ist demnach ein Datenformat um diese Tripel abzuspeichern. Damit die Knoten eindeutig identifiziert werden können, werden sie durch URIs dargestellt, wobei das Subjekt und das Prädikat immer eine URI sein müssen und das Objekt ein Datenformat, auch Literal genannt, also eine Zahl oder eine Zeichenkette, sein kann. URI bedeutet Uniform Resource Identifier und ist eine Zeichen-

¹⁰² Vgl. Prodromou (2008).

¹⁰³ Vgl. Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 17.

¹⁰⁴ <http://www.w3.org/TR/rdf-syntax-grammar/>, letzter Zugriff am 16.07.2009.

¹⁰⁵ Eigene Darstellung in Anlehnung an Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 41.

folge. Damit kann eine abstrakte oder physikalische Ressource eindeutig dargestellt werden. Es gibt zwei bekannte Arten von URIs: Zum einen die URL (Uniform Resource Locator) zum adressieren von Web Ressourcen und zum anderen URN (Uniform Resource Name) um andere Ressourcen wie ISBN zu identifizieren. RDF nutzt immer http URIs, wobei die URIs nicht real existieren sollten. Denn wenn eine URI in der Form `http://www.example.org/SemanticWeb` angezeigt wird, ist es nicht ersichtlich, ob damit der Begriff Semantic Web oder die konkrete Seite gemeint ist. Daher sollte die URI bevorzugt in der Form: `„http://www.example.org/Main#SemanticWeb“` angegeben werden. Eine gültige RDF Struktur hat beispielsweise die folgende Form (wobei „rdf“ für den Namensraum `„http://w3.org/1999/02/22-rdf-syntax-ns#“` steht und „ex“ eine Kurzschreibweise von `„http://example.org/“` ist):

```
<rdf:Description rdf:about="http://example.org/Jaguar">
  <ex:istAutoVon>
    <rdf: Description rdf:about="http://example.org/Bob">
      </rdf:Description>
    </ex:istAutoVon>
  </rdf:Description>106
```

Es ist anzumerken, dass die Abkürzungen der Namensräume wie „rdf“ und „ex“ nicht in den Attributen wie „about“ verwendet werden dürfen.

Zudem bietet RDF noch einige weitere grundlegende Definitionen. So können Literale als bestimmte Formate wie eine Zahl, eine Zeichenkette oder Datum definiert werden. Die in RDF enthaltenen Container erlauben es mehrere Objekte, wie in unserem Beispiel „Bob“ eines ist, zusammen zu fassen, um sie auf diese Weise einfacher zu notieren. Durch einen leeren Knoten werden diese Container mit dem Rest des Graphen verbunden. Das Auto Jaguar könnte zum Beispiel nicht nur „Bob“, sondern auch den Personen „Tim“ und „John“ gehören. Dabei verweist „istAutoVon“ auf den leeren Knoten, welcher dann auf die Objekte „Bob“, „Tim“ und „John“ zeigt. Nur Objekt oder Subjekt können leere Knoten sein, welche zur Identifizierung IDs (eine eindeutige Nummer) bekommen. Diese Knoten sind erforderlich um n-äre¹⁰⁷ Relationen herzustellen. Binäre¹⁰⁸ Relationen können in manchen Fällen zu Missverständnissen führen, wie folgendes Beispiel zeigt:

Subjekt	Prädikat	Objekt
Rezept	Zutat	Mehl .

¹⁰⁶ Eigene Darstellung in Anlehnung an Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 41.

¹⁰⁷ N-äre Relation bezeichnet die Verbindung von beliebig vielen Mengen miteinander vgl. Olbrich (1998), S. 7.

¹⁰⁸ Eine binäre Relationen ist die Verbindung von genau zwei Objekten, vgl. Olbrich (1998), S. 7.

Rezept	Zutat	Zucker .
Rezept	Menge	200g .
Rezept	Menge	450g .

Tabelle 2: Missverständnisse in binären Graphen¹⁰⁹

Durch die reine binäre Zuordnung ist nicht klar, welche Zutat in welcher Menge benötigt wird. Neben der Lösung dieses Problems bieten Container zudem die Möglichkeit anzugeben, ob folgende Elemente in einer geordneten oder ungeordneten Menge gelistet sind oder ob sie alternative Möglichkeiten aufzeigen.

Ein weiteres Ausdrucksmittel in RDF sind Collections, welche ähnlich wie Container mehrere Objekte gruppieren, mit der Ausnahme, dass Collections eine abgeschlossene Menge bilden. Als Hilfsmittel werden auch hier leere Knoten verwendet. Die Prädikate eines RDF Tripels werden mit „rdf:first“ und „rdf:rest“ benannt, wodurch eine Menge immer aus einem ersten Objekt und dem Rest besteht und der Rest wiederum aus einem ersten Objekt und dem Rest besteht. Die letzte Menge hat als die leere Liste als Rest und wird durch die URI „rdf:nil“ repräsentiert.

Um Wiederholungen zu vermeiden, können so genannte Vokabulare definiert werden, in denen die Struktur, in der die Daten als RDF-Tripel gespeichert werden sollen, vorgegeben ist. Eines der bekanntesten thematischen Vokabulare ist FOAF, kurz für Friend of a Friend, zur Darstellung von Benutzern in sozialen Netzwerken und deren Verbindungen untereinander. Um jedoch komplexere Netzwerke darzustellen benötigt es differenzierterer Beschreibungstechniken.¹¹⁰

RDFS

Das RDF Schema, kurz RDFS, ist ein Vokabular welches seit dem Jahr 2004 als Empfehlung des W3C gilt. Es erweitert RDF um einige Ausdrücke und beschreibt nicht nur ein Thema. RDFS basiert auf der Syntax von RDF und gilt als leichtgewichtige Ontologiesprache. Die Grenzen wurden hier sehr eng gesetzt im Gegensatz zu einer schwergewichtigen Ontologiesprache wie OWL.

RDFS bietet die Möglichkeit Ressourcen einer bestimmten Klasse, das heißt einer Menge von Elementen, zuzuordnen. Dies wird mit dem allgemeinen Prädikat „rdf:type“ erreicht. Dadurch können Objekte als Über- oder Untergeordnet deklariert werden. Folgendes Beispiel zeigt dies (die Abkürzung „ex“ steht wieder für den Namensraum „http://example.org/“ und „rdfs“ für den Namensraum „http://w3.org/2000/01/rdf-schema#“):

¹⁰⁹ Eigene Darstellung in Anlehnung an Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 55f.

¹¹⁰ Vgl. Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 36ff,

Vgl. Grigoris/ Harmelen (2008), S. 65ff.

Subjekt	Prädikat	Objekt
ex:Jaguar	rdfs:type	ex:Tier .
ex:Jaguar	rdfs:subClassOf	ex:Säugetier .
ex:Säugetier	rdfs:subClassOf	ex:Lebewesen .

Tabelle 3: RDFS Klassen¹¹¹

Diese Schreibweise wird als Turtle bezeichnet und ermöglicht als (inoffizielle) RDF Syntax einen Tripel kompakt und übersichtlich darzustellen. Sie basiert auf der von Tim Berners-Lee vorgeschlagenen „Notation 3“. Diese Zuordnung ermöglicht es mittels anderer Techniken, wie dem Reasoner FaCT++¹¹², Schlussfolgerungen zu ziehen. Durch Transitivität ist zu erkennen, dass Jaguar auch eine Unterklasse von Lebewesen ist. Diese Zusammenhangsstrukturen lassen sich auch für Prädikate definieren. Zum Beispiel wäre die Eigenschaft „istNichte“ eine Oberklasse von „istEinzigeNichte“ und wird folgendermaßen definiert:

Subjekt	Prädikat	Objekt
ex:istEinzigeNichte	rdfs:subPropertyOf	ex:istNichte .
ex:Alice	ex:istEinzigeNichte	ex:Bob .

Tabelle 4: RDFS Zusammenhänge für Prädikate¹¹³

Wenn Alice die einzige Nichte von Bob ist, trifft natürlich auch die Aussage zu, dass sie seine Nichte ist. Auch erlaubt RDFS durch die Bereichsangabe „rdfs:range“ bestimmte Wertebereiche zu definieren. Beispielsweise sollte das Alter einer Person nicht negativ sein:

Subjekt	Prädikat	Objekt
ex:hatAlter	rdfs:range	xsd:nonNegativInteger .

Tabelle 5: Restriktionen durch RDFS definieren¹¹⁴

Hier wird deutlich, dass RDFS wie RDF auf XML und XML Schema aufbaut und somit auch die dort existierenden Definitionen nutzt. RDFS hat noch weitere Eigenschaften, durch die es sich von RDF abhebt, auf die hier nicht näher eingegangen wird. Trotzdem hat auch RDFS seine Grenzen,

¹¹¹ Eigene Darstellung in Anlehnung an Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 70f.

¹¹² <http://owl.man.ac.uk/factplusplus/>, letzter Zugriff am 16.07.2009.

¹¹³ Eigene Darstellung in Anlehnung an Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 75.

die erreicht werden, wenn versucht wird komplexere Zusammenhänge zu modellieren. Es ist nicht möglich Negationen, Inversitäten oder Rollen auszudrücken. Einen Ausdruck wie „istFrau“ als das Gegenteil von „istMann“ darzustellen bedarf zusätzlicher Definitionen.¹¹⁵

OWL

Web Ontology Language, kurz OWL, ist eine komplexe Sprache zur Beschreibung von Ontologien und wird seit 2004 vom W3C empfohlen. Warum OWL (englisch: Eule) nicht WOL abgekürzt wird, darüber gibt es mehrere Spekulationen. Möglich ist, dass sich OWL besser ausspricht und durch die Bezeichnung Eule mit Weisheit in Verbindung gebracht werden soll. Es gibt für OWL zwei Syntaxen: Die eine basiert ebenfalls auf der RDF-Syntax, die andere heißt abstrakte OWL Syntax. Letztere ist für den Menschen leichter zu lesen, wurde aber nur für OWL Full und OWL DL verfügbar gemacht. OWL ist so komplex und ausdrucksstark, dass beschlossen wurde es in drei Abstufungen einzuteilen, OWL Full, OWL DL und OWL Lite. Die einzelnen Abstufungen haben die folgenden Eigenschaften:

OWL Full	<ul style="list-style-type: none"> - enthält OWL DL und OWL Lite - enthält als einzige OWL-Teilsprache ganz RDFS - sehr ausdrucksstark - Semantik enthält einige Aspekte, die aus logischem Blickwinkel problematisch sind - unentscheidbar - wird durch aktuelle Softwarewerkzeuge nur bedingt unterstützt
OWL DL	<ul style="list-style-type: none"> - enthält OWL Lite und ist Teilsprache von OWL Full - entscheidbar - wird von aktuellen Softwarewerkzeugen fast vollständig unterstützt - Komplexität NexpTime (worst-case)
OWL Lite	<ul style="list-style-type: none"> - ist Teilsprache von OWL DL und OWL Full - entscheidbar - weniger ausdrucksstark - Komplexität ExpTime (worst-case)

Tabelle 6: Unterschiede in den OWL Versionen¹¹⁶

¹¹⁴ Eigene Darstellung in Anlehnung an Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 75.

¹¹⁵ Vgl. Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 66ff.

Die Vorteile von OWL DL gegenüber OWL Full sind im Wesentlichen, dass es entscheidbar ist und somit die Frage, ob eine Aussage aus einer Ontologie gefolgert werden kann, mit einem Algorithmus beantwortbar, der stets terminiert. Daher ist es leichter entsprechende Werkzeuge hierfür zu entwickeln. Allerdings ist OWL DL nicht voll kompatibel zu RDFS, da die Typen „`rdfs:class`“ und „`rdfs:property`“ nicht verwendet werden dürfen. Auch müssen die Typen wie Klassen, Instanzen und Rollen strikt getrennt werden. Das Rollenmodell unterliegt außerdem noch weiteren Restriktionen. Der Vorteil den OWL Lite bietet ist, dass es wegen der geringen Komplexität für den Benutzer wesentlich verständlicher ist und auf Grund seiner Einfachheit auch für die Entwickler von entsprechenden Werkzeugen zur Umsetzung leichter verwendet werden kann. Auf Grund seiner starken Einschränkungen, ist es in der Praxis jedoch von untergeordneter Bedeutung.¹¹⁷ Die Einschränkungen enthalten alle Beschränkungen von OWL DL und zusätzlich noch Restriktionen wie in einigen Fällen vorgeschriebene Klassennamen oder Rollenrestriktionen sowie Zahlenrestriktionen, welche sind auf Binärwerte beschränkt sind.

Doch welche neuen Vorteile bringt OWL mit sich? Von OWL Full ausgehend, existiert das erwähnte Rollenkonzept für komplexere Klassen. Rollen sind OWL-Property's und sind in abstrakter und konkreter Form vorhanden. Abstrakte Rollen verbinden zwei Individuen, konkrete Rollen ein Individuum mit einem Datenwert. Unter Individuen sind RDF-Instanzen von Klassen und Datenwerten, welches Elemente aus Datentypen wie Zeichenketten oder Zahlen sind, zu verstehen. Mit Rollen ist es möglich beispielsweise mittels „`owl:inverseOf`“ eine Ausschlussbeziehung wie „`istChef`“ zu „`hatChef`“ auszudrücken. Für Individuen gibt es eine Reihe von Beziehungen wie „`owl:sameAs`“ für Gleichartigkeit, „`owl:differentFrom`“ für Verschiedenheit und viele weitere. Dadurch lässt sich einfach beschreiben, dass zum Beispiel „`Sofa`“ und „`Couch`“ die gleiche Bedeutung haben. Wenn `Sofa` eine Unterklasse von `Wohnzimmermöbel` ist, folgt daraus durch Inferenz, dass auch `Couch` eine Unterklasse von `Wohnzimmermöbel` ist. Die Stärke von OWL liegt außerdem im Einsatz der logischen Konstruktoren „`und`“, „`oder`“ und „`nicht`“, die in OWL mit „`owl:intersectionOf`“, „`owl:unionOf`“, „`owl:complementOf`“ beschrieben werden. Dadurch können komplexe logische Konstrukte modelliert werden, welche hingegen in OWL Lite nur eingeschränkt genutzt werden können.

Zur Modellierung von OWL gibt es verschiedene Werkzeuge. Protegé ist einer der bekanntesten Ontologieeditoren für diesen Bereich. Die Wichtigkeit der Aufteilung von OWL in drei Abstufungen wird deutlich, wenn die OWL-Inferenzmaschinen betrachtet werden. In diesen gibt es nur sehr wenige die überhaupt OWL DL komplett unterstützen. OWL wird heute zumeist im Intranet zur Wissensorganisation eingesetzt. Die große Vision für das Internet ist, dass die Ontologien dort frei zugänglich sind und dadurch miteinander verbunden werden können. Der Zusammenschluss würde zu einem riesigen Datenpool führen. Es gibt jedoch gerade auf diesem Gebiet, dem so genannten OWL mapping oder OWL merging noch viele Probleme und großen Forschungsbedarf.¹¹⁸ Denn

¹¹⁶ Vgl. Hitzler, P./ Krötzsch, M./ Rudolph, S./ Sure, Y. (2008). S. 127.

¹¹⁷ Vgl. Grigoris / Harmelen (2008), S. 118.

¹¹⁸ Vgl. ebd., S. 219ff, 245ff.

auch wenn zwei Ontologien das Selbe bezeichnen, unterscheiden sie sich in ihrer Spezifikation und Konzeptionalisierung.¹¹⁹

SPARQL

Nachdem herausgestellt wurde, dass Daten in Ontologien wie OWL organisiert und im RDF Format abgespeichert werden können, muss nun geklärt werden, welche Möglichkeiten bestehen diese Daten wieder auszulesen. Dabei ist es nicht unbedingt von Bedeutung alle Instanzen auslesen zu wollen, was zu einer unüberschaubaren Datenmenge führen würde. Wichtiger ist eine definierte Anzahl auszulesen oder eine bestimmte Menge an Elementen heraus zu filtern. Um diese Aufgaben kümmert sich die Abfragesprache SPARQL, welches eine rekursive Abkürzung für SPARQL Protocol and RDF Query Language ist. Diese Sprache ist erst seit Anfang 2008 eine W3C Empfehlung, wodurch die Aktualität und der Fortschritt der Entwicklung dieses Standards deutlich werden. SPARQL enthält neben der Abfragesprache für RDF-Graphen auch ein SPARQL-Protokoll zur Anfragenübermittlung sowie die SPARQL-Kodierung von Ergebnissen. Der Fokus dieses Abschnitts wird auf der Betrachtung der Abfragesprache liegen. SPARQL ist eine an SQL angelehnte Sprache und besteht in seiner einfachsten Form aus einem „SELECT“ und einem „WHERE“ Teil:

```
PREFIX ex: <http://example.org/>
```

```
SELECT ?Jaguar ?Lebensraum
```

```
WHERE
```

```
{  ?tier    ex:Gattung <http://test.de/Saugetier> .  
    ?tier    ex:TierArt  ?Jaguar .  
    ?tier    ex:Kontinent  ?Lebensraum . }
```

¹²⁰

Dabei zeigt das Fragezeichen im Abfragekode, dass es sich hier um eine Variable handelt. Statt des Fragezeichens ist es ebenfalls erlaubt ein „\$“ zu verwenden. Durch „PREFIX“ können die URIs in der Abfrage abgekürzt werden, in dem dort Namensräume definiert werden. Als Ergebnis wird eine Tabelle zurückgeliefert, in diesem Fall mit den zwei Spalten, „Jaguar“ und „Lebensraum“. Die Ergebnisausgabe wird durch die Angaben im SELECT-Teil bestimmt. Neben diesen einfachen Abfragen erlaubt es SPARQL zusätzlich gruppierende Graph-Muster abzufragen. Das bedeutet, die Abfrage kann im WHERE-Teil durch „{ }“ zusammengefasst werden um somit komplexere Abfragen zu generieren. Dies geschieht mit Hilfe von Befehlen wie „OPTIONAL“, welcher zusätzliche Ergebnisse zulässt, UNION, welcher zwei Muster verknüpft und vieler weiterer.

Auch können bestimmte Werte abgefragt werden, in dem zum Beispiel statt „?Lebensraum“ direkt nach „Afrika“ gefragt wird. Ebenso gibt es Filtermöglichkeiten, die es zulassen, Schranken nach

¹¹⁹ Vgl. Hitzler / Krötzsch / Rudolph / Sure (2008), S. 125ff.

oben oder unten zu setzt. In unserem Beispiel könnte mit dem Befehl „FILTER (?alter <5)“ im WHERE Teil das Alter der ausgegebenen Tiere begrenzt werden. Da die Größe der Lösungsmenge, auf Grund dezentraler Speicherung im Web, unbekannt sein kann, gibt es zusätzlich die Möglichkeit die Ergebnismenge mit LIMIT zu begrenzen oder mit OFFSET den Zählstart festzulegen.

SPARQL ist eine Abfragesprache für RDF-Graphen. Jedoch bilden RDFS oder OWL nicht nur einfache Graphen. Von OWL DL ausgehend beschreibt dies nicht nur eine einzige, sondern viele mögliche Interpretationen die unendlich groß sein können. Um diese Probleme zu umgehen bietet SPARQL konjunktive Abfragen an. Diese ähneln logischen Formeln aus der Prädikatenlogik. Es gibt hierzu keinen offiziellen Standard oder eine normierte Syntax, jedoch eine Vielzahl von Implementierungen.¹²¹

RDFa

Daten im Web können verteilt sein, doch wäre hierbei die Frage zu stellen, in welcher Form sie existieren. Im einfachsten Fall können die RDF oder OWL Dateien frei zugänglich und damit auslesbar sein. Zum anderen gibt es aber auch die Möglichkeit die Daten in HTML zu kennzeichnen. So werden die Informationen einerseits für den Menschen auf der Website lesbar, andererseits auch für Maschinen verständlich. Dies geschieht, ähnlich wie bei microformats, durch Auszeichnungen in der HTML Sprache, in diesem Fall mittels RDFa. Doch statt für jedes Szenario eine Abbildungsvorschrift zu definieren, wird hier eine generelle Vorschrift definiert. Durch Namensräume können mehrere Abbildungsvorschriften gleichzeitig verwendet werden. Als Datenformat wird das einheitliche RDF genutzt, weshalb es ausreichend ist einen einzigen Extraktor zu erstellen, der die Daten ausliest und als RDF-Graph speichert. Der Code sieht, für das folgende FOAF-Beispiel, folgendermaßen aus:

```
<div xmlns:foaf="http://xmlns.com/foaf/0.1/" about="#me" rel="foaf:knows">
  <ul>
    <li typeof="foaf:Person">
      <a property="foaf:name" rel="foaf:homepage" href="http://example.com/bob">Bob</a>
    </li>
    <li typeof="foaf:Person">
      <a property="foaf:name" rel="foaf:homepage" href="http://example.com/eve">Eve</a>
    </li>
    <li typeof="foaf:Person">
```

¹²⁰ Eigene Darstellung, angelehnt an Hitzler / Krötzsch / Rudolph / Sure (2008), S. 202.

¹²¹ Vgl. Hitzler / Krötzsch / Rudolph / Sure (2008), S. 202ff, 233ff.


```

    <a property="foaf:name" rel="foaf:homepage" href="http://example.com/manu">Manu</a>
  </li>
</ul>
</div>122

```

Hierbei wird ein weiterer wesentlicher Unterschied zu microformats deutlich. RDFa benutzt nicht nur Attribute wie „rel“ und „rev“, sondern auch „typeof“, „about“ oder „property“ und einige mehr, welche im aktuellen XHTML 1.0 Standard noch nicht enthalten sind. Diese werden erst in der nächsten Version XHTML 2.0 hinzukommen, welche sich zurzeit noch in der Entwicklung befindet. Damit RDFa schon jetzt einsetzbar ist und bei der Validierung erkannt wird, wurde der neue Doctype XHTML+RDFa eingeführt. Die Verwandtschaft zu RDF wird durch das Tripel deutlichen, welches durch das „about“, die „property“ und den Wert des Tags mit dem Attribut „property“ definiert ist. Für das obige Beispiel wären das für den ersten Eintrag die Werte „#me“, „foaf:name“ und „Bob“.¹²³

GRDDL

Um RDFa wieder aus einer XHTML Seite herauszulesen wird ein Extraktor benötigt. Ein solcher vom W3C empfohlener ist Gleaning Resource Descriptions from Dialects of Languages, kurz GRDDL. Dieser nimmt eine Transformation des Dokumentes anhand einer XSLT Datei vor, welche im Kopf des Dokumentes angegeben sein muss. Ein GRDDL-aware agent liest das Dokument, erkennt die Transformationsvorschrift und wandelt die Daten in gültiges RDF um. Von diesem Format aus können die Daten in verschiedenen Programmen verwendet werden.



Abb. 6: GRDDL im Einsatz¹²⁴

Dies funktioniert sowohl mit RDFa als auch mit microformats. Bei letzterem muss jedoch für jedes Format eine neue Transformationsvorschrift erstellt werden. Hier wird einer der Nachteile eines uneinheitlichen Systems deutlich.¹²⁵

¹²² Vgl. Adida/ Birbeck (2008).

¹²³ Vgl. Blumauer/ Pellegrini (2009), S 153ff.

¹²⁴ Quelle: http://www.w3.org/TR/grddl-scenarios/select_item.png, letzter Zugriff am 16.07.2009.

¹²⁵ Vgl. Adida/ Ayers/ et al. (2007).

SWRL

Der nächste Baustein in der Übersichtsgrafik aus Abb. 7 des W3C ist Logic und Proof. Hierfür gibt es SWRL, welches auf OWL aufbaut und um wichtige Funktionen ergänzt, die es erlauben komplexere Schlussfolgerungen abzuleiten. Ebenfalls basiert es auf der Rule Markup Language (kurz RuleML)¹²⁶, welches ein auf XML basierendes Regelwerk zur Beschreibung von Transformation, insbesondere von semantischen Daten, ist. Allerdings kann RuleML an sich in der aktuellen Version keine RDF Daten transformieren, weshalb es mit RuleML Lite eine Version gibt die kompatibel zu SWRL ist. Die Semantic Web Rule Language¹²⁷ ist seit 2004 nur ein Vorschlag und noch kein offizieller Standard des W3C. Allerdings ermöglicht SWRL die Definition von Horn-Klauseln ähnlichen Regeln. Darunter sind Klauseln der Prädikatenlogik zu verstehen, die höchstens ein negatives Atom besitzen. Es lassen sich also Regeln in der Form „Aus A und B folgt es gilt C“ formulieren. Für Horn-Klauseln¹²⁸ lässt sich in polynomieller Zeit bestimmen ob sie lösbar sind auch wenn SWRL an sich unentscheidbar ist.¹²⁹

Zusammenfassung und Ausblick

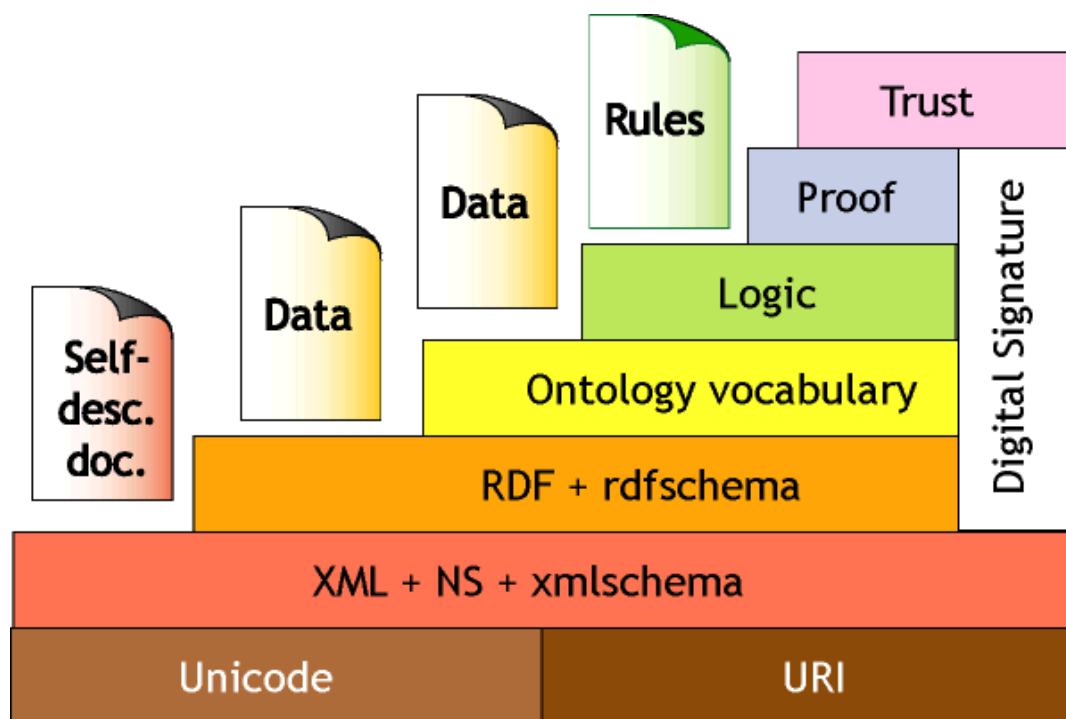


Abb. 7: Technologien für das semantic Web¹³⁰

¹²⁶ www.ruleml.org, letzter Zugriff am 16.07.2009.

Vgl. Grigoris/ Harmelen (2008), S. 171ff.

¹²⁷ www.w3.org/Submission/SWRL, letzter Zugriff am 16.07.2009.

¹²⁸ Vgl. Wing Ng (1993), S. 399ff.

¹²⁹ Vgl. Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 252.

¹³⁰ Quelle: <http://www.w3.org/2000/Talks/1206-xml2k-tbl/sweb-stack.gif>, letzter Zugriff am 16.07.2009.

Neben den hier besprochenen Techniken existieren noch viele weitere wie eRDF, kurz für embedded RDF, welches es ermöglicht RDF mit normalen (X)HTML Befehlen in das Dokument einzubinden. Für den Ansatz eines ontologiebasierenden Taggingsystems sind alle benötigten Formate vorgestellt worden, wie die Vor- und Nachteile des Taggings und zum anderen die Möglichkeit durch Ontologien eine Ordnung zu schaffen aus der strukturierte Informationen abgeleitet werden können. Als nächstes müssen die Risiken betrachtet werden, die solche Systeme in sich bergen können.

3.2.3 Sicherheit

Das Semantic Web eröffnet durch die automatische Erschließung der Bedeutung von Inhalten eine große Anzahl an Möglichkeiten. Allerdings entstehen auf der anderen Seite neue Risiken. Diese sind auf den ersten Blick nicht unbedingt erkennbar, denn auch wenn die Inhalte semantisch annotiert werden, so sind sie trotzdem schon jetzt im Internet verfügbar.

Datenschutz

Das Risiko besteht in der Zusammenführung von Daten aus verschiedenen Quellen. Denn durch das Sammeln von Daten kann aus vielen Einzelinformationen aus dem Web ein neues Gesamtbild entstehen.¹³¹ Beispielsweise könnte ein Nutzer auf einer Vereinsseite seine Hobbys darstellen, darunter vielleicht auch Extremsportarten. Auf einer anderen Internetseite präsentiert er sich für seinen Arbeitgeber. Diese zwei Profile können verschiedene Daten beinhalten, da der Arbeitgeber nicht alles aus dem Privatleben des Arbeitnehmers wissen soll. Doch Agenten im Semantic Web können diese Daten sammeln und zu einem einzigen umfangreichen Profil ergänzen. Die Selbstbestimmung über die Daten ist nicht mehr gegeben. Es geht daher nicht darum, dass eine Person die persönlichen Daten nicht preisgeben soll, denn dies geschieht bereits jetzt und ist eine Frage der Gesellschaftsform.¹³² Hierbei geht es um den Schutz der einzelnen Daten. Dieser darf hingegen auch nicht zu restriktiv sein, denn ein Datenaustausch für geschäftliche Zwecke wie im E-commerce muss weiterhin problemlos möglich sein. Es müssten so genannte „Sphären“ erstellt werden, in denen die Daten genutzt werden dürfen und es muss der Kontext erhoben werden, in dem die Daten angegeben wurden.¹³³

Zurzeit ist im Semantic Web nicht nachvollziehbar woher die Informationen kommen, da sie von beliebigen Quellen geladen werden. Die gesammelten Daten können außerdem auf Servern liegen, die im Ausland stehen und demnach nicht nationalem Recht untergeordnet sind. Eine Beeinflussung der eigenen Daten könnte so erheblich erschwert werden.¹³⁴

¹³¹ Vgl. Kim/ Hoffman/ Martin (2002).

¹³² Vgl. Blumauer/ Pellegrini (2009), S 485.

¹³³ Vgl. Semantic Web School (2006).

Web of Trust und digitale Signaturen

Für einen Agenten ist es ein Problem, wenn die gesammelten Daten auf Servern im Ausland liegen, denn woher soll er wissen, welche Daten vertrauenswürdig und richtig sind?¹³⁵ Auch im Semantic Web kann jeder Benutzer Inhalte publizieren, ohne dass es eine zentrale Instanz zur Qualitätskontrolle gibt. Daher wird für Agenten ein Rating System benötigt, mit dessen Hilfe die Verlässlichkeit der Informationen ermittelt werden kann. Mit Web of Trust und den digitalen Signaturen werden die letzten noch offenen Bausteine besprochen, die in der Grafik des W3C zum Aufbau von Semantik existierten. Eine Möglichkeit wäre die Aggregation von unterschiedlichen Bewertungen aus unterschiedlichen Quellen zu einem einheitlichen Ranking, sowie die Abschätzung der Qualität der Bewertungen. Ein weiterer Aspekt ist, dass nicht alle Daten von jedem Agenten ausgelesen werden sollen. Beispielsweise sollte in einer Ontologie eines Krankenhauses der Agent des Lieferanten für Medikamente die Anzahl der Neubestellungen auslesen können, jedoch darf er nicht in Erfahrung bringen können für wen die Medikamente sind. Der Agent eines Hausarztes hingegen sollte Zugriff auf die Patientendaten dieses Arztes haben. Um dies zu gewährleisten können mittels einer XML Encryption einzelne Inhalte einer RDF Datei verschlüsselt werden und mit XML Signature sicher gestellt werden, dass der Absender nicht gefälscht wurde. Mit einem öffentlichen und einem privaten Schlüssel kann validiert werden, dass nur die berechtigte Person die Daten liest.¹³⁶ Das W3C hat in diesem Zuge den Platform for Privacy Preferences Standard, kurz P3P, verabschiedet. Dieser soll die Privatsphäre schützen, indem bestimmte Inhalte einer Website maschinen- und menschenlesbar gekennzeichnet werden können. Jedoch ist dies nur eine Kennzeichnung der Inhalte, so dass Agenten, die P3P nicht unterstützen oder böswillige sind, nicht daran gehindert werden die Informationen auszulesen.¹³⁷

Zusammenfassung und Ausblick

Zurzeit, da sich das Semantic Web noch in der Entwicklung befindet, werden vornehmlich die Vorteile der neuen Technik wahrgenommen. Die Nachteile werden sich erst im Laufe der Zeit bemerkbar machen, sind aber jetzt zum Teil schon auszumachen. Insbesondere der Schutz der Privatsphäre könnte in diesem Zuge weiter vermindert werden. Da die geplanten Ontologien für das Lernportal learnuzz nicht öffentlich gemacht werden und auch die personenbezogenen Daten in kennwortgeschützten Bereichen liegen, bestehen für dieses Portal die gelisteten Risiken nicht, der Datenschutz ist gewährleistet. Auch die Inhalte, die zur Ermittlung von Zusammenhängen ausgelesen werden, stammen aus dem eigenen Datenbestand und sind somit vertrauenswürdig, weshalb die Probleme des Web of Trust ebenfalls nicht auf diese Arbeit zutreffen.

¹³⁴ Vgl. Bettel (2007).

¹³⁵ Vgl. O'Hara (2004).

¹³⁶ Vgl. Kim/ Hoffman/ Martin (2002).

¹³⁷ Vgl. Cranor/ Langheinrich/ et al. (2002).

4 TECHNISCHE UMSETZUNG

Die Recherchen der letzten Kapitel haben ergeben, dass sich für die Umsetzung des Projektes auf der Plattform learnuzz das Prinzip des Taggens als sinnvoll herausstellt. Die Plattform basiert, wie in der Einleitung bereits beschrieben, auf dem Web 2.0 Prinzip: User generated Content. Benutzer haben die Möglichkeit eigenständig Inhalte einzustellen und diese direkt mit Tags zu versehen. Allerdings konnte im dritten Kapitel auch dargestellt werden, dass Tagging einige Probleme mit sich bringt. Zum Beispiel erfordert die Erkenntnis, dass der Tag „BWL“ in Verbindung zum Tag „BWL A“ steht, beziehungsweise das „BWL A“ eine konkretere Bezeichnung für ein getaggttes Objekt ist, semantische Techniken. In der Umsetzung soll das Prinzip des Taggens daher mittels RDF und OWL um eine Ontologie erweitert werden.¹³⁸ Damit die Metadaten¹³⁹ nicht unstrukturiert vorliegen, können sie in einem Thesaurus Management redaktionell gepflegt werden. Dies ist bei wachsenden Systemen der zurzeit empfehlenswerteste Weg ein kontrolliertes Vokabular zu erstellen.¹⁴⁰

Das Ziel dieser Arbeit ist nicht das Anlegen konkreter OWL Daten, denn diese werden mit Hilfe von Werkzeugen wie einem Ontologieeditor erzeugt.¹⁴¹ Stattdessen sollen geeignete Werkzeuge gefunden werden, mit denen sich ein Empfehlungssystem erstellen lässt und welche den Aufwand sowohl für den Benutzer als auch für den Administrator und Plattformbetreiber weitestgehend minimieren. Die Art der eingesetzten Tools ist stark von der Arbeitsumgebung abhängig. Für die Plattform learnuzz wird keine RDF optimierte Datenbank wie Oracle 10g¹⁴² eingesetzt sondern eine einfache MySQL-Datenbank¹⁴³. Learnuzz wurde unter dem Einsatz des Frameworks „Zend“ entwickelt. Um die zuvor beschriebenen semantischen Technologien in eine Webseite zu integrieren, gibt es ebenfalls verschiedene Werkzeuge.¹⁴⁴ Für PHP ist dies ARC2: „ARC is a flexible RDF system for semantic web and PHP practitioners“. (<http://arc.semsol.org/>, letzter Zugriff am 16.07.2009)

4.1 Vokabulare

Neben den eingesetzten Werkzeugen ist die Auswahl von Abbildungsvorschriften, so genannten Vokabularen, einer der wichtigsten Grundsteine. Um eine Ontologie zu verwenden werden Vokabulare, welche die Struktur vorgeben, benötigt. Unter Vokabularen ist die Zusammenstellung von Bezeichnern mit klar abgegrenzter Bedeutung wie das RDF Vokabular selbst zu verstehen. Ihr

¹³⁸ Vgl. Blumauer/ Pellegrini (2009), S 227ff,

Vgl. Gaiser (2008), S. 194ff.

¹³⁹ Metadaten sind Daten über Daten, vgl. Hitzler / Krötzsch / Rudolph / Sure (2008) S. 17.

¹⁴⁰ Vgl. DCMI (2005),

Vgl. Gaiser (2008), S. 238.

¹⁴¹ Vgl. Breitman/ Casanova/ Truszkowski (2007), S. 203.

¹⁴² Vgl. ebd., S. 211.

¹⁴³ MySQL ist eine frei verfügbare Datenbank, <http://www.mysql.de/>, letzter Zugriff am 16.07.2009.

Einsatz ist auch in der Darstellung von Informationen zu finden. In Anwendungsfällen, wo durch Ontologien Ableitungsregeln definiert werden, liegt der Schwerpunkt bei Vokabularen auf der Sprache und deren syntaktischen Regeln. Es gibt Vokabulare für die verschiedensten Themen.¹⁴⁵

Eines der bekanntesten Vokabulare ist FOAF, kurz für Friend of a Friend.¹⁴⁶ Mit diesem können Personen, Gruppen oder Organisationen beschrieben werden. Auch Verknüpfungen zwischen Personen sind durch die Eigenschaft „foaf:know“ möglich, so dass sich hierdurch Netzwerke bilden lassen. Mit den verschiedenen anderen Eigenschaften (Properties) wie „foaf:interest“ für die Interessen einer Person oder „foaf:publication“ für Veröffentlichungen steht ein breites Spektrum zur Abdeckung der häufigsten Anwendungsfälle bereit. Da ein Vokabular nie alle möglichen Fälle abdecken kann, ist FOAF so interoperabel, dass es als Grundlage für Erweiterungen genutzt werden kann, wie zum Beispiel für FOAFRealm. Dies erweitert FOAF um Funktionalitäten wie Authentifizierung oder Zugriffskontrolle.¹⁴⁷ Ein weiteres Vokabular ist SIOC, kurz für „Semantically-Interlinked Online Communities“.¹⁴⁸ Es beschreibt die Hauptkonzepte einer Online-Community und ermöglicht die Diskussion über Plattformen hinweg.¹⁴⁹ Dieses dient zum Informationsaustausch internetbasierter Diskussionsforen und Community Portalen. Die Entwickler von FOAF, SIOC und SKOS, einem Vokabular zur Wissensorganisation, haben sich zusammengeschlossen, um ein gemeinsames Konzept zur Zusammenarbeit der Ontologien zu erstellen. Dieses soll Konflikte und Duplikate vermeiden und ist ein derzeit einzigartiges Vorgehen.¹⁵⁰

Für das Lernportal ist, neben FOAF, ein weiteres Vokabular von Interesse, welches sich in der Mitte der drei verbündeten Vokabulare sieht.¹⁵¹ Social Semantic Cloud of Tags¹⁵², kurz SCOT, ist eine Abbildungsvorschrift um Tag-Events und Folksonomies auszudrücken.¹⁵³ Es beinhaltet die Verwaltung von Tags, Tag Clouds, die Taggingaktivität an sich und eine besonders für die Bewertung von Tags wichtige Funktion um die Stärke des Zusammenhangs zu messen. Primär wurde SCOT für Tag Clouds entworfen und wird als Mittler zwischen den verschiedenen Vokabularen SKOS für Beschreibungen, SIOC für Communities und FOAF für Freundesnetzwerke, genutzt wie Abb. 8 verdeutlicht.

¹⁴⁴ <http://semanticweb.org/wiki/Tools>, letzter Zugriff am 16.07.2009.

¹⁴⁵ Vgl. Hitzler/ Krötzsch/ Rudolph/ Sure (2008), S. 48.

¹⁴⁶ <http://www.foaf-project.org/>, letzter Zugriff am 16.07.2009.

¹⁴⁷ Vgl. Blumauer/ Pellegrini (2009), S. 188f.

¹⁴⁸ Vgl. ebd., S. 189,

<http://www.sioc-project.org>, letzter Zugriff am 16.07.2009.

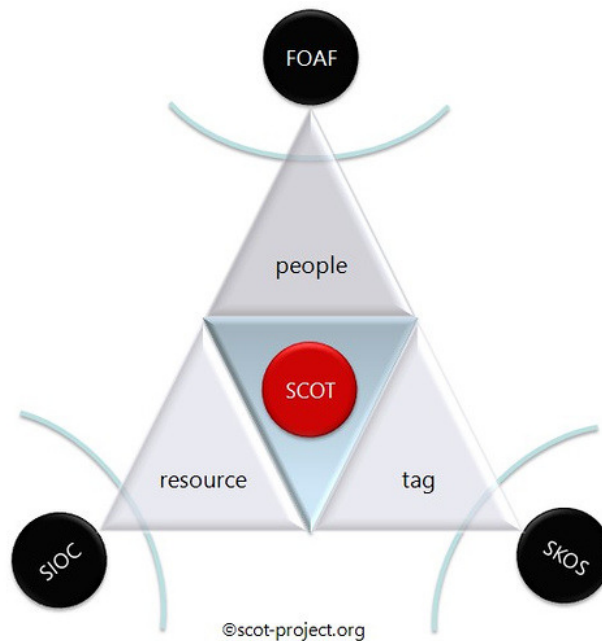
¹⁴⁹ Vgl. Blumauer/ Pellegrini (2009), S. 222.

¹⁵⁰ Vgl. ebd., S. 234.

¹⁵¹ Vgl. ebd., S. 234.

¹⁵² <http://scot-project.org/>, letzter Zugriff am 16.07.2009.

¹⁵³ Vgl. Gaiser (2008), S. 234.

Abb. 8: Verbindung von SCOT zu anderen Vokabularen¹⁵⁴

SCOT wurde nach dem Prinzip „never repeat yourself“ erstellt, was bedeutet, dass bevorzugt bereits vorhandene Vokabulare benutzt werden anstatt neue Konstrukte zu entwerfen. Daher enthält es als Grundlage das Konzept von Richard Newman’s Tag Ontology¹⁵⁵.

4.1.1 Dublin Core – semantische Grunddateninformation

Dublin Core Metadata Initiative¹⁵⁶, kurz DCMI, ist ein einfacher Standard um Ressourcen im Internet zu beschreiben und wird von vielen weiteren Vokabularen als Basis genutzt. Als ältestes Vokabular entstand es 1995 auf einem Workshop in Dublin. Es dient dazu Dokumente im Web durch Metadaten wie Autor, Betreff, Datum und weitere zu beschreiben. Dieses Vokabular wird von der DCMI Organisation, einer nicht-kommerziellen Vereinigung, geführt und gehört nicht zum W3C. Dabei konzentriert sich DCMI auf die grundlegenden Informationen, die für die meisten Ressourcen passen. Einige häufig verwendete Ausdrücke sind:¹⁵⁷

Name	Definition
dc:Title	Beschreibt den Namen der Ressource. Ein Beispiel wäre „Always on my mind“.

¹⁵⁴ Quelle: http://farm1.static.flickr.com/174/446172158_595486ac6c.jpg, letzter Zugriff am 16.07.2009.

¹⁵⁵ Vgl. Newman (2005).

¹⁵⁶ <http://dublincore.org/>, letzter Zugriff am 16.07.2009.

¹⁵⁷ Vgl. Frodl/ Fischer/ et al. (2007).

dc:Subjekt	Das Thema der Ressource. Ein Beispiel wäre „Music“.
dc:Description	Eine Beschreibung der Ressource. Ein Beispiel wäre „Dies ist ein berühmtes Lied des Musikers Elvis Presley“.
dc:Type	Art oder Gattung der Ressource. Ein Beispiel wäre „Sound“.
dc:Source	Eine verwandte Ressource, von der die beschriebene Ressource abgeleitet ist. Ein Beispiel wäre „Bild vom Cover der Album CD“.
dc:Date	Ein Zeitpunkt oder eine Zeitspanne im Zusammenhang mit einem Ereignis im Entwicklungsprozess der Ressource. Ein Beispiel wäre „1935-01-08“.
dc:Format	Das Dateiformat, der Datenträger oder der Umfang der Ressource. Ein Beispiel wäre „image/gif“ oder „23 kB“.

Tabelle 7: Häufig genutzte Dublin Core Auszeichnungen¹⁵⁸

Die gesamte Liste befindet sich in deutscher Übersetzung unter http://www.kimforum.org/material/pdf/uebersetzung_dcmes_20070822.pdf (letzter Zugriff am 16.07.2009) im Internet.¹⁵⁹

4.1.2 SKOS – Beschreibung von Wissensressourcen

Das zuvor schon erwähnte und weit verbreitete Vokabular Simple Knowledge Organisation System, kurz SKOS ist eine Spezifikation von allgemeinen Konzepten und Bezeichnungen.¹⁶⁰ Es ist ein W3C Vokabular¹⁶¹ und befindet sich zurzeit noch in der Entwicklung. SKOS soll die Beschreibung von Wissens-Organisationssystemen in einer für Maschinen verständlichen Sprache ermöglichen, da viele Taxonomien, Thesauern und Klassifikationen in einer recht ähnlichen Struktur vorhanden sind. Es erlaubt bevorzugte oder alternative Bezeichnungen in verschiedenen Sprachen. SKOS ist ein sehr umfangreiches Vokabular, so dass an dieser Stelle nur einige wichtige Elemente vorgestellt werden:

„skos:Concept“ ist eine generelle Klasse welche eine konzeptionelle Ressource beschreibt. Sie ist absichtlich verallgemeinert, um konkrete Ausprägungen von ihr abzuleiten.

¹⁵⁸ <http://purl.org/dc/terms/>, letzter Zugriff am 16.07.2009,
<http://dublincore.org/>, letzter Zugriff am 16.07.2009.

¹⁵⁹ Vgl. DCMI (2008),
Vgl. Manola/ Miller (2004),
Vgl. Frodl/ Fischer/ et al. (2007).

¹⁶⁰ Vgl. Mikhalenko (2005),
Vgl. Isaac/ Summers (2009).

¹⁶¹ <http://www.w3.org/TR/skos-reference/>, letzter Zugriff am 16.07.2009.

„skos:Collection“ ist ebenfalls eine Klasse, welche Ansammlungen Begriffen mit Bedeutungszusammenhang beinhaltet. Es gibt einige Abwandlungen wie zum Beispiel „skos:OrderedCollection“. Diese steht für eine geordnete Ansammlung von Konzepten.¹⁶²

Mit „skos:prefLabel“ wird die bevorzugte Bezeichnung der Ressource angegeben. Dazu können mittels „skos:altLabel“ alternative Bezeichnungen hinzugefügt werden. Diese dürfen, müssen aber nicht, Synonyme der bevorzugten Bezeichnung sein. Auch miteinander in Verbindung stehende Worte wie „skos:prefLabel“ = „gut“, „skos:altLabel“ = „schlecht“ sind möglich, ebenso verschiedene Sprachen der Bezeichnung.

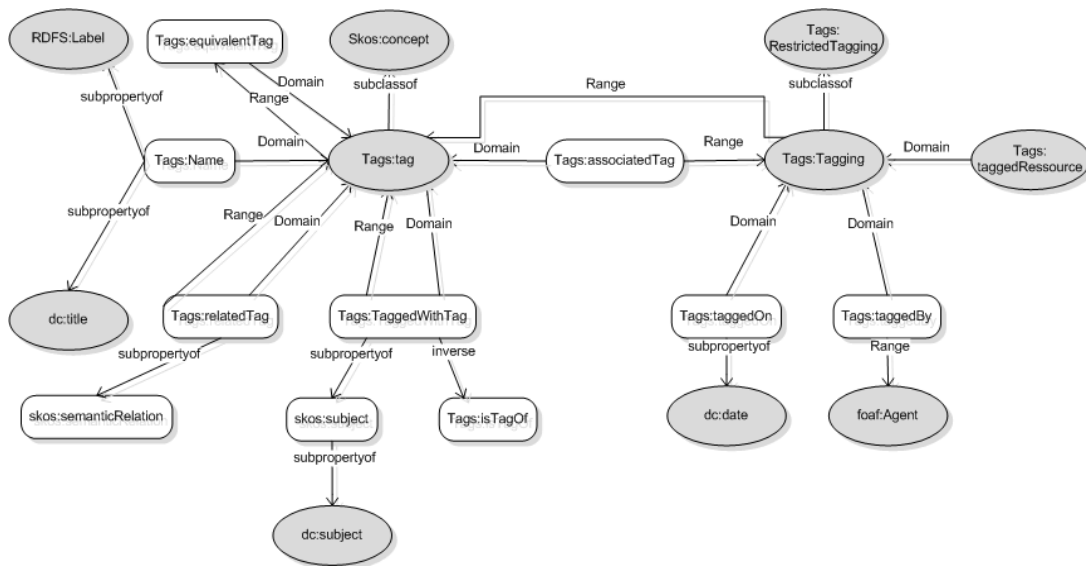
In SKOS gibt es acht verschiedene Möglichkeiten eine Ressource zu beschreiben. Allgemein durch „skos:note“ welches sich dann in die Spezialfälle „skos:publicNote“ und „skos:privateNote“ für öffentliche oder private Dokumente, „skos:definition“ für eine komplette Erklärung der Ressource, „skos:scopeNote“ für eine Kurzbeschreibung der wichtigsten Inhalte, „skos:example“ für ein Beispiel, „skos:historyNote“ für wesentliche Änderungen in der Ressource, „skos:editorialNote“ für administratorische Anmerkungen und „skos:changeNote“ für kleinere Änderungsanmerkungen. Mit dieser Klassifikation können die Beschreibungen sehr präzise unterschieden werden.

Ebenso können Beziehungen zwischen Objekten mit SKOS modelliert werden. Dazu gibt es die allgemeine Beschreibung „skos:semanticRelation“ und ihre Kinder „skos:broader“, „skos:narrower“ und „skos:related“. Die Eigenschaft Broader beschreibt eine übergeordnete Bezeichnung, während Narrow auf eine spezifischere Bezeichnung zeigt. Mit Hilfe dieser beiden inversen Eigenschaften können zum Beispiel Hierarchien erzeugt werden. Für assoziative Beziehungen wird die Eigenschaft „Related“ verwendet.

4.1.3 Tag – Beschreibung von Tags

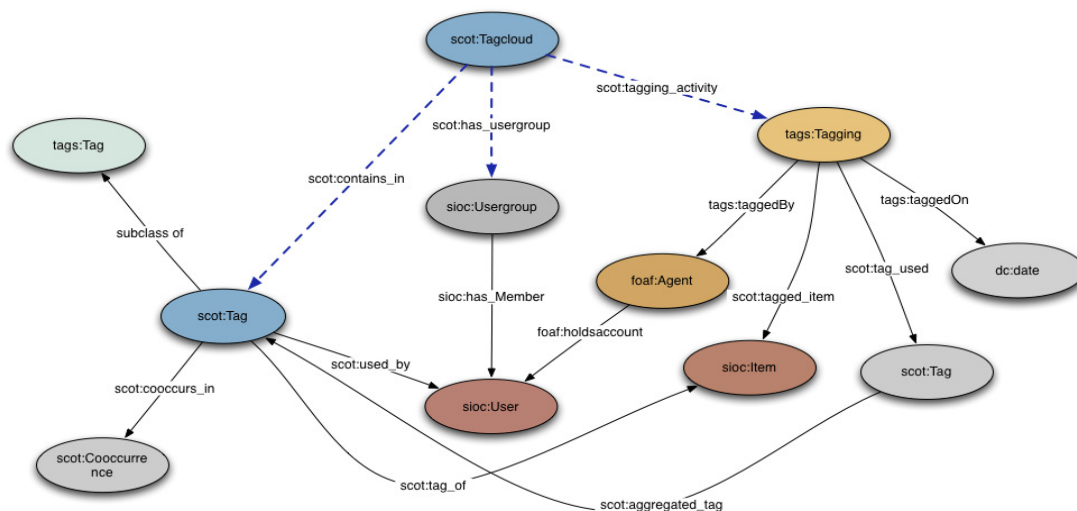
Die Richard Newman's Tag Ontology dient dazu einen Tag im Detail zu beschreiben und zu definieren. Abb. 9 veranschaulicht dies grafisch. Sie wurde 2004 von Richard Newman entworfen und greift auf das Konzept von SKOS zurück, in dem es einen Tag als Unterklasse von „skos:Concept“ deklariert. Ein Tag wird hier durch verschiedene Attribute beschrieben. Zum einen hat ein Tag eine Bezeichnung („tags:Name“), welche er von dem Vokabular Dublin Core und dessen Beschreibung „dc:title“ erbt. Tags können auf verschiedene Art und Weise in Verbindung stehen. Auf der einen Seite ist eine „Gleichheit“ möglich, die durch „tags:equivalentTag“ ausgedrückt wird, auf der anderen Seite können sie in einer hierarchischen Beziehung stehen. Dies wird durch „tags:relatedTag“ ausgedrückt, welches von „skos:semanticRelation“ erbt. Neben dem Tag selbst hat diese Ontologie ein zweites Konzept, die Aktion des taggens, welche sich „tags:Tagging“ nennt. Mit diesem Entwurf wird gespeichert, welche Ressource von einem bestimmten Benutzer zu einer eindeutigen Zeit getagt wurde. Das Datum leitet sich von dem Dublin Core Vokabular ab und nutzt „dc:date“, während der Benutzer ein „foaf:Agent“ ist. Damit baut bereits diese kleine Ontologie auf drei existierenden Vokabularen auf.

¹⁶² <http://www.w3.org/TR/2005/WD-swbp-skos-core-spec-20051102/>, letzter Zugriff am 16.07.2009.

Abb. 9: Die Tag Ontologie als Graph¹⁶³

4.1.4 SCOT – erweiterte Tag und Tag Cloud Beschreibung

SCOT¹⁶⁴ macht von anderen Vokabularen noch größeren Gebrauch als Richard Newman's Tag Ontology, weshalb sich daraus eine wesentlich komplexere Ontologie ergibt. Eine vereinfachte Darstellung ist in Abb. 10 zu sehen. Im Wesentlichen besteht SCOT aus den folgenden vier Bereichen.

Abb. 10: Graphische Darstellung der SCOT Ontologie¹⁶⁵

¹⁶³ Quelle: Eigene Darstellung, siehe Anhang A.1 für eine vergrößerte Ansicht.

¹⁶⁴ Vgl. Kim/ Breslin / et al. (2008).

¹⁶⁵ Quelle: http://farm3.static.flickr.com/2073/2316114955_e820385ffe_o.png, letzter Zugriff am 16.07.2009, siehe Anhang A.2 für eine vergrößerte Ansicht.

Zum einen existiert die Tag Cloud, welche als Sammlung von Tags dient. Daneben gibt es die einzelnen Tags mit ihren Eigenschaften und auch die von Richard Newman's Tag Ontologie bekannte Tagging Aktivität. Dazu ist noch die Cooccurrence vorhanden, mit der sich ermitteln lässt wie oft Tags zusammen verwendet werden. Insbesondere mit Blick auf ein Empfehlungssystem, in dem von Tags ausgehend andere geeignete Objekte gefunden werden sollen, ist dies eine hilfreiche Information. Die vier genannten Unterbereiche werden nun im Detail dargestellt.

Tag Cloud

Abb. 11 zeigt die Struktur der Tag Cloud. Diese ist einfach aufgebaut, da sie nur einige Datenfelder enthält, aber kaum komplexe Anordnungen. Auch das Konstrukt der Tag Cloud wurde nicht neu erfunden, sondern leitet sich aus dem Container des SIOC Vokabulars ab. Angehängt werden der Cloud die Tags, wobei der Verweis sowohl mittels des Trippels über den Tag mit „scot:Tag scot:contained_in scot:Tagcloud“ erfolgen kann, als auch umgekehrt mit „scot:Tag scot:contains scot:Tagcloud“. Diese inversen Möglichkeiten werden bei vielen Elementen geboten. Ebenfalls zur Wolke gehört die für ein Social Network nützliche Benutzergruppe, abgeleitet von „sio:Usergroup“ sowie der Ersteller der Cloud, der „foaf:Agent“. Dazu wird die Tag Cloud über die Verbindung „tags:Tagging“ mit der Tagging Aktivität in Bezug gesetzt sowie das Datum der letzten Änderung gespeichert. Zur statistischen Auswertung kann die absolute Anzahl der Tags, der Items und der Zusammenhänge direkt als Zahl gespeichert werden. Der Begriff Items beschreibt Objekte wie die einzelnen Gruppen oder Fragen in diesem System.

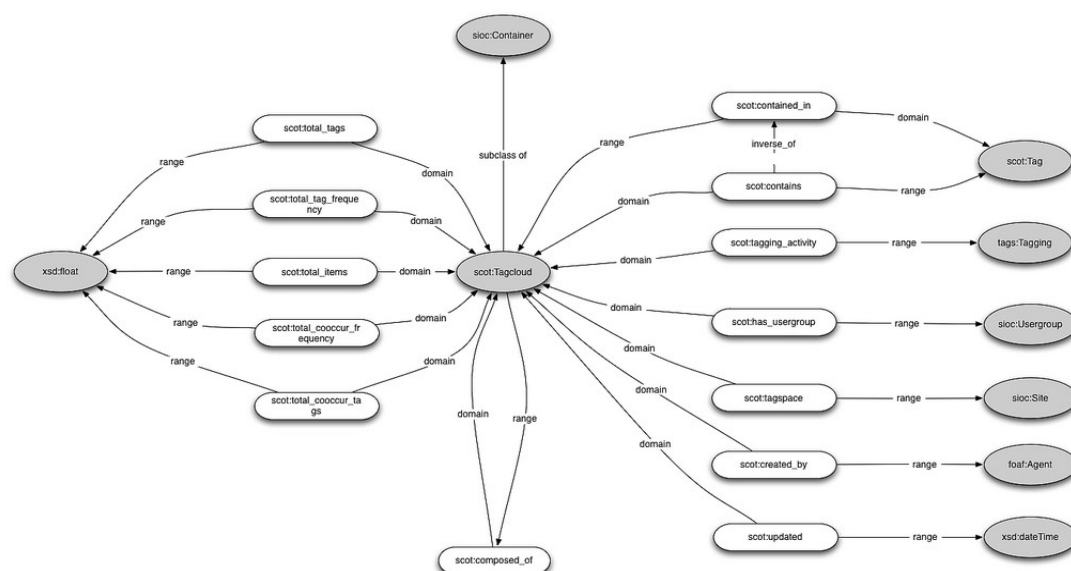


Abb. 11: Schematische Darstellung der SCOT Tag Cloud mit ihren Eigenschaften¹⁶⁶

¹⁶⁶ Quelle: http://farm4.static.flickr.com/3183/2316922726_03cef3c089.jpg, letzter Zugriff am 16.07.2009, siehe Anhang A.3 für eine vergrößerte Ansicht

Tag

Im Gegensatz zur Tag Cloud ist der Ontologieteil zum Bereich Tag umfangreicher und komplexer wie anhand von Abb. 12 zu erkennen ist. Der Tag („scot:Tag“) ist eine Unterklasse des Tags aus der Tag Ontology von Richard Newman („tags:Tag“), welche wiederum eine Unterklasse von „skos:Concept“ des Vokabulars SKOT ist. Als wichtigstes Kriterium ist die Verbundenheit zum getaggten Objekt (Item) beziehungsweise zum taggenden Benutzer zu erwähnen. Das Item wird über „scot:has_tag“ mit dem Tag verbunden, beziehungsweise in entgegen gesetzter Richtung über „scot:tag_of“ vom Item zum Tag. Die Beziehung „scot:tag_of“ wiederum ist eine Unterklasse von „tags:is_TagOf“ der Tag Ontology und erbt dessen Eigenschaften. Der Benutzer wird über die Klasse „sioc:User“ angegeben und als direkte URL zum Benutzerprofil gespeichert sowie auch das Datum, wann der Tag zuletzt verwendet wurde und der Name des Tags. Die Häufigkeit der Verwendung des Tags wird über die Beziehungen „scot:ownAFrequency“ und „scot:ownRFrequency“ angegeben und als Zahl gespeichert. Das Prädikat „scot:ownAFrequency“ bedeutet dabei die absolute Häufigkeit des Tags, beispielhaft wird ein Tag achtmal verwendet. Im Gegensatz dazu speichert „scot:ownRFrequency“ die relative Häufigkeit des Tags in der Tag-Cloud.

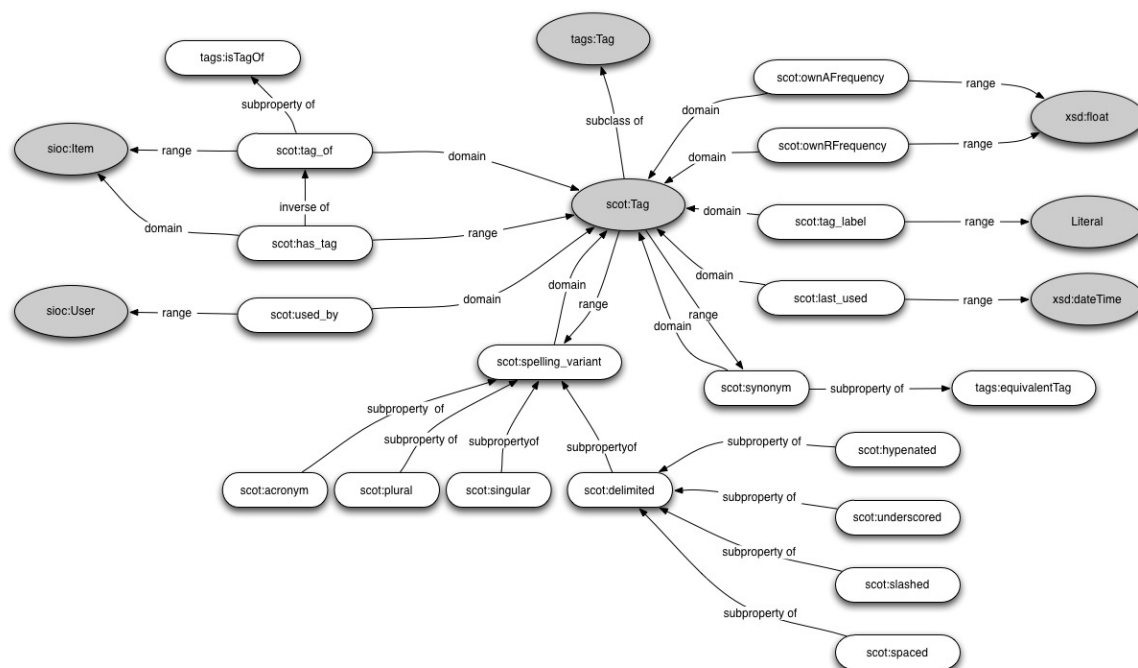


Abb. 12: Schematische Darstellung eines SCOT Tag mit seinen Eigenschaften¹⁶⁷

Eine Eigenschaft von Tags ist, dass sie nicht nur eine Zeichenkette darstellen, sondern für den Menschen eine bestimmte Bedeutung haben. Daher kann es zwischen verschiedenen Schlagwörtern Verbindungen geben, welche in einem Empfehlungssystem erkannt werden sollten. Die erste Mög-

¹⁶⁷ Quelle: http://farm3.static.flickr.com/2169/2316922774_80870ce81f.jpg, letzter Zugriff am 16.07.2009, siehe Anhang A.4 für eine vergrößerte Ansicht.

lichkeit besteht darin Tags mit gleicher Bedeutung zu definieren. Dies kann über die Verbindung „scot:synonym“, welches eine Unterklasse von „tags:equivalentTag“ der Tag Ontology ist, geschehen. Noch vielfältiger ist die Möglichkeit verschiedene Schreibweisen auszudrücken. Allgemein geschieht dies durch das Prädikat der Sprachvariante („scot:spelling_variant“), allerdings wird hier die Möglichkeit geboten gezielt zu unterscheiden, ob es sich dabei um ein Akronym („scot:Aacronym“), die Pluralform („scot:plural“), die Singularform („scot:singular“) oder um einen aus mehreren Worten bestehenden Begriff („scot:delimited“) handelt. Bei dem letzteren wird zusätzlich differenziert, wie die Trennung erfolgt. Dies kann über einen Bindestrich („scot:hypenated“), einen Unterstrich („scot:underscored“), einen Schrägstrich („scot:slashed“) oder ein Leerzeichen („scot:spaced“) herausgestellt werden. In der Umsetzung eines Empfehlungssystems wird sich darauf beschränkt Tags mit ähnlicher Bedeutung durch „scot:spelling_variant“, zu verbinden. Eine weitere Unterteilung wird hier nicht zu einem besseren Vorschlag führen, da die Tatsache, dass die Tags in Verbindung stehen, eine ausreichende Information darstellt, welche nicht weiter verfeinert werden muss.

Es gibt allerdings durch die Verbindung der Tags Klasse („scot:Tag“) mit der Tag Ontology von Richard Newman („tags:tag“) noch eine weitere übergeordnete Möglichkeit Verbindungen zwischen Tags darzustellen. Die Klasse „tags:tag“ hat die Eigenschaft „tags:relatedTag“, welche eine Untereigenschaft der Beziehung des SKOT Vokabulars „skos:semanticRelation“ ist. Dabei hat „skos:semanticRelation“ die Unterklassen „skos:broaderTransitive“ und „skos:narrowerTransitive“, welche wiederum die Untereigenschaft „skos:broader“ beziehungsweise „skos:narrower“ besitzen. Mit diesen Eigenschaften lassen sich Hierarchien erzeugen. „Broader“ steht dabei für eine allgemeinere Bezeichnung, also im weitesten Verständnis für eine höhere Hierarchieebene und „Narrower“ steht für spezifisch, also für eine tiefere Hierarchieebene. Mit diesen Verbindungen ist es möglich baumähnliche Strukturen in der Ontologie zu definieren, welche hier in einem Vokabularmanager genutzt werden sollen. Dort wird das Prädikat „skos:broader“ benutzt um Oberklassen von Tags zu definieren und „skos:narrower“ für Unterklassen. Insgesamt bietet diese Ontologie die Möglichkeit Tags wie „Vorlesung“ und „BWL A“ in eine Hierarchie zu gliedern, als auch „BWL A“, „BWL_A“ und „bwlA“ als gleichbedeutend zu kennzeichnen.

Cooccurrency

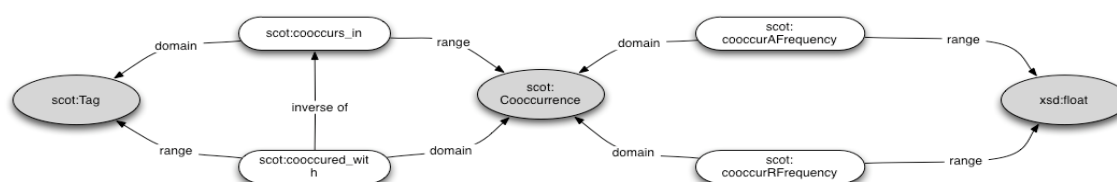


Abb. 13: Schemantische Darstellung der Zusammenhangsstärke zwischen Tags in SCOT¹⁶⁸

¹⁶⁸ Quelle: http://farm3.static.flickr.com/2371/2316922806_707b6b41fd_o.png, letzter Zugriff am 16.07.2009, siehe Anhang A.5 für eine vergrößerte Ansicht.

Einen sehr einfachen, aber doch wichtigen Teil der Ontologie ist der Graph über die Zusammenhänge („scot:cooccurrence“). Damit werden Tags, die zusammen auftreten, also zum Beispiel gemeinsam mit einem Objekt getaggt wurden, als zusammenhängend erkannt und abgespeichert. Dabei besteht die Klasse aus einer eindeutigen Zeichenkette, welche über die ein Prädikat („scot:cooccurs_in“) mit den dazugehörigen Tags verbunden wird. Außerdem lässt sich mit „scot:cooccurrenceAFrequency“ und „scot:cooccurrenceRFrequency“ wieder die absolute und relative Häufigkeit dieser Zusammenhänge erfassen. Eine Interpretation dieser Zusammenhänge wäre, dass ein Objekt, welches mit Tag A annotiert wurde, und ein Objekt, welches mit Tag B ausgezeichnet wurde in Zusammenhang stehen, wenn beide Tags zu der gleichen Zusammenhangesklasse („scot:cooccurrence“) Klasse gehören.

Tagging Aktivität

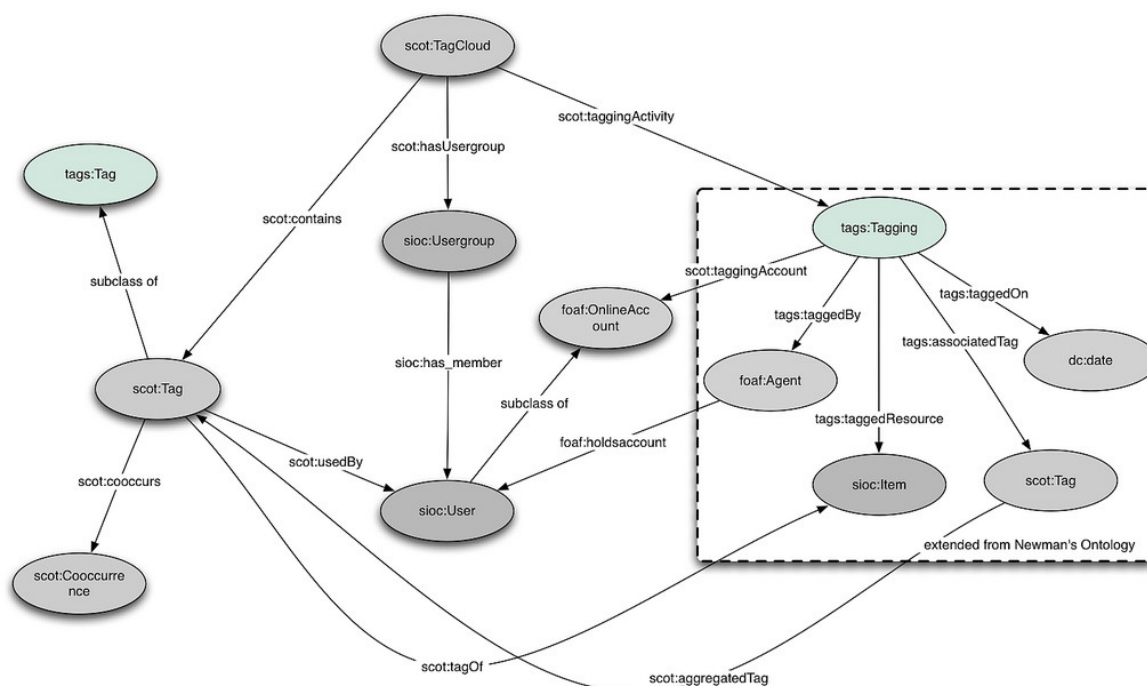


Abb. 14: Graphische Darstellung der SCOT Ontologie¹⁶⁹

Als letzter Teil der Ontologie wäre dann noch die Tagging Aktivität zu beschreiben, welche in dem markierten Kasten in Abb. 14 zu sehen ist. Zu dieser wird, wie in Richard Newman's Tag Ontology, das getaggte Objekt („sioc:Item“) sowie der Benutzer („foaf:Agent“), der Tag („scot:Tag“) und das Datum gespeichert. Dazu wird mit dem Prädikat „scot:taggingAccount“ auf ein Online Profil („foaf:OnlineAccount“) verwiesen, welches eine Oberklasse des Benutzers („sioc:user“) der Tag Cloud ist. Mittels „tags:associatedTag“ wird dieses Konstrukt mit der Tag Cloud verbunden.

¹⁶⁹ Quelle: http://farm4.static.flickr.com/3265/2610870382_5cb795988e.jpg, letzter Zugriff am 16.07.2009, siehe Anhang A.6 für eine vergrößerte Ansicht.

4.2 Werkzeuge

Nachdem nun mit der Ontologie das theoretische Modell definiert wurde, auf dem das Empfehlungssystem aufbauen wird, müssen nun verschiedene Werkzeuge gesucht werden, mit denen das System realisiert werden soll. Als erstes erfolgt eine Systemanalyse um festzustellen, welche Restriktionen existieren.

4.2.1 Zend Framework

Das bestehende Portal baut auf dem Framework „Zend“¹⁷⁰ auf, welches die erste Vorgabe ist. Unter framework.zend.com lässt sich dieses Framework herunterladen. Es ist eine freie Software und steht unter der BSD Lizenz. Das System setzt auf der Programmiersprache PHP auf und bietet vorgefertigte Schnittstellen zu einer MySQL Datenbank. Die Dokumentation ist sehr ausführlich und enthält ein Tutorial¹⁷¹ für den Einstieg. Da der Prototyp der Software in dieses System eingebettet werden muss, wird zur weiteren Einarbeitung in die Applikationslogik eine einfache Gruppenverwaltung eingebaut. Sie soll das Erstellen und Auflösen von Gruppen erlauben sowie das Ein- und Austreten in diese. Das Framework „Zend“ ist modular strukturiert und beinhaltet einen großen Pool an vorgegebenen Bibliotheken. Durch die Modularität ist es jedoch einfach eigene Bibliotheken in das System einzubinden beziehungsweise mit Bestehenden auszutauschen.

4.2.2 ARC2

Mit dem Einsatz des Frameworks „Zend“ folgt, dass die learnuzz Applikation auf PHP und einer MySQL Datenbank aufbaut. Für diese Datenbank wird ein Adapter gesucht, welcher die RDF Logik implementiert, da die bestehende Umsetzung die Standardvorgabe nutzt. ARC2¹⁷² ist ein solches System. Es implementiert eine auf MySQL aufgebaute Datenstruktur und erlaubt das Speichern und Auslesen von RDF-Tripeln in diese. Dies geschieht mittels der SPARQL Anfragesprache, welche vom System in SQL umgewandelt und verarbeitet wird. Die erste Version wurde von Benjamin Nowack im Jahr 2006 veröffentlicht. Die aktuelle Version 2 wird von der Firma Semsol (semsol.com) gesponsert. Auf Grund der geringen Größe des Projektes fehlt es der Dokumentation an Genauigkeit. Es gibt ein Tutorial, welches alle Funktionen kurz erläutert, jedoch ist der Beispielcode sehr gering, weshalb es einige Zeit des Testens bedarf, bis die Logik verständlich ist. Auch wird ein sehr gutes Verständnis von RDF und insbesondere SPARQL vorausgesetzt. Dafür gibt es einen sehr guten Support seitens des Entwicklers, welcher über eine Mailingliste abläuft. Auf Verbesserungsvorschläge wird zeitnah eingegangen und entdeckte Fehler werden sehr schnell behoben. Auch existieren mehrere Projekte, die bereits auf dieser Bibliothek aufbauen. Erwähnenswert ist hier insbesondere ein Modul für das Content Management System Drupal, welches das

¹⁷⁰ <http://framework.zend.com/>, letzter Zugriff am 16.07.2009.

¹⁷¹ <http://framework.zend.com/docs/quickstart>, letzter Zugriff am 16.07.2009.

¹⁷² <http://arc.semsol.org/>, letzter Zugriff am 16.07.2009.

Einbinden des schon erwähnten Services Calais ermöglicht.¹⁷³ Auch der Service von LODr¹⁷⁴ baut auf diesem System auf. Er ermöglicht es die Tags mit dem Service Linked Data¹⁷⁵ zu verbinden. Das besondere hierbei ist, dass der Quelltext frei verfügbar ist. Denn insgesamt ist es schwer Quelltext-Beispiele für den Einsatz von ARC2 zu finden.

4.2.3 Ontologieeditoren

Als nächstes soll die Idee aufgegriffen werden ein Expertensystem umzusetzen, in dem es möglich ist die Tags zu ordnen. Durch einen solchen Vokabularmanager sollen sie in Ober- und Unterklassen und in gleichbedeutende Tags eingeteilt werden können.

Protegé

Zur Umsetzung scheint sich auf den ersten Blick der Ontologieeditor Protegé¹⁷⁶ zu eignen. Es ist ein auf Java basierendes Werkzeug, mit dem sich Ontologien wie SCOT kreieren lassen. Zurzeit befindet sich der Editor in einer Umbruchphase.¹⁷⁷ Die alte Version 3.x wird durch eine überarbeitete Version 4 ersetzt, die wesentliche Unterschiede aufweist. Einige Unterschiede beruhen darauf, dass gewisse Teile wie der Frames Editor noch nicht für die neue Version umgesetzt wurden. Dazu zählen auch viele Plug-Ins, welche bis zum heutigen Zeitpunkt nur für die Version 3.x existieren als auch die Unterstützung der Abfragesprache SPARQL. In Version 3.x gab es eine Datenbankbindung über JDBC. Dies ist eine auf Java-basierende Datenbankschnittstelle für relationale Datenbanken. Auch diese ist noch nicht in die neue Version umgesetzt worden. Auf der anderen Seite legt die neue Version andere Schwerpunkte. Sie ist ein reines OWL Framework und bietet keine RDF(S) Unterstützung an. Dafür wurden bereits einige Reasoner wie FaCT++ oder Pellet¹⁷⁸ implementiert, welche geschwindigkeitsoptimiert wurden. Ein Reasoner ist ein Algorithmus der in einer Ontologie automatisch logische Zusammenhänge erkennt und einbaut. Wenn ein Element A invers zu einem Element B ist, erkennt der Reasoner, dass auch alle Unterelemente von A und B invers sein müssen. Damit nimmt er dem Menschen nicht nur Arbeit ab, er erkennt auch komplexe fehlerhafte Zuordnungen, die zu einem Widerspruch führen können. Auffällig ist außerdem die überarbeitete Benutzeroberfläche. Die Dokumentation ist sehr ausführlich, auch wenn es hier durch die Umstellung auf die neue Version 4.0 Komplikationen gibt. Die Anleitungen sind entweder für Version 3 geschrieben, berücksichtigen aber nicht die letzten Änderungen, oder sind für eine frühe Version 4 entworfen worden.

¹⁷³ <http://drupal.org/project/opencalais>, letzter Zugriff am 16.07.2009.

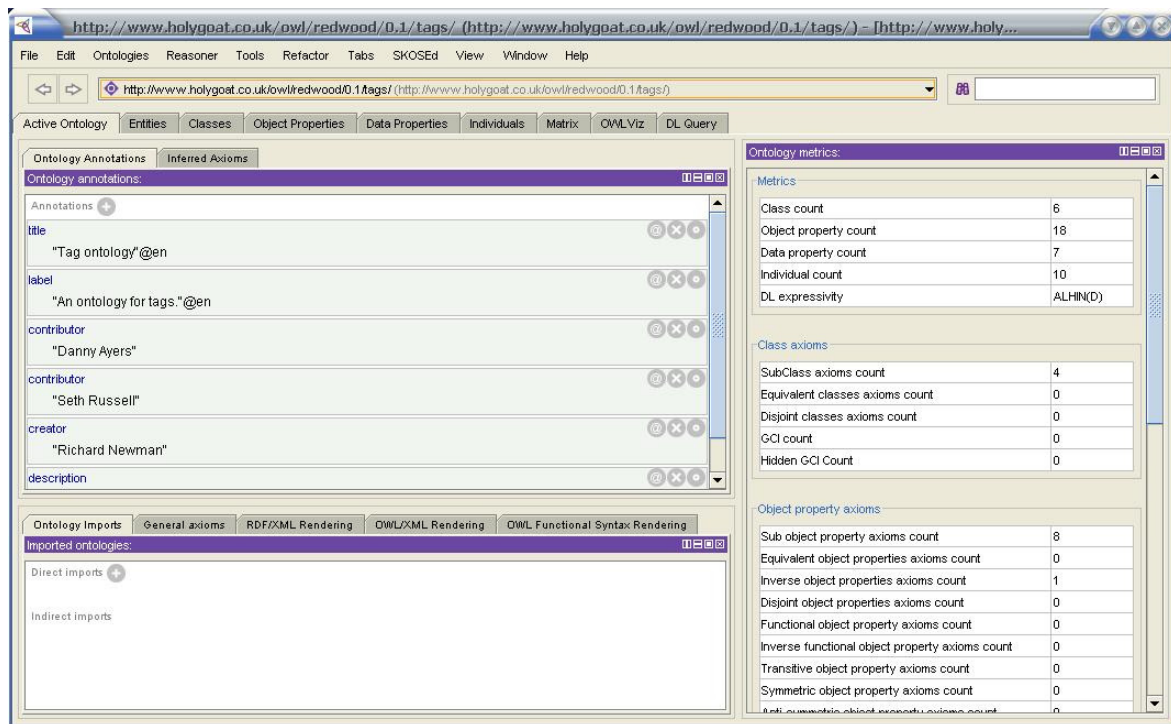
¹⁷⁴ <http://lodr.info/>, letzter Zugriff am 16.07.2009.

¹⁷⁵ <http://linkeddata.org/>, letzter Zugriff am 16.07.2009.

¹⁷⁶ <http://protege.stanford.edu/>, letzter Zugriff am 16.07.2009.

¹⁷⁷ <http://protegewiki.stanford.edu/index.php/Protege4Migration>, letzter Zugriff am 16.07.2009.

¹⁷⁸ <http://clarkparsia.com/pellet>, letzter Zugriff am 16.07.2009.

Abb. 15: Ansicht einer Ontologie im Protege Editor¹⁷⁹

Nach der Einarbeitungsphase trat ein wesentliches Problem auf: Der Ontologieeditor ist für die Verwendbarkeit als Vokabularmanager nicht nur überdimensioniert, das Auslesen der Daten ist auf RDF-Dateien oder große Datenbanken ausgelegt, jedoch nicht auf das MySQL basierte ARC2 System. Da für PHP und MySQL zurzeit kein anderes System als ARC2 empfohlen werden kann, hat dieser Editor keinen Nutzen für das zu entwickelnde System.

Thesaurus Manager

Ein anderes Tool ist der Thesaurus Manager.¹⁸⁰ Es ist ein Werkzeug zur Erstellung von SKOS Vokabularen. Dieses Tool basiert ebenfalls auf Java und hat das gleiche Problem wie Protegé, es kann nicht auf eine ARC2 Datenbank zugreifen. Jedoch ist es übersichtlich und bietet die gewünschte Funktion Tags hierarchisch einzuteilen, da diese Einteilung über das SKOS Vokabular erfolgt. Daher ist dieses Programm für eine Orientierung hilfreich.

Da noch kein Editor für das zu entwickelnde System mit den Restriktionen existiert, wird der nächste Schritt sein Anforderungen zu definieren und einen eigenen Editor zu entwerfen. Eine andere Möglichkeit, wie das Entwickeln eines Wrappers, welcher die Daten aus der MySQL Datenbank ausliest und dem entsprechenden Editor zur Verfügung stellt, ist auf Grund der wenigen Anforderungen an unseren gewünschten Vokabularmanager keine vergleichbare Option. Der Aufwand inklusive der Einarbeitung würde das Zeitlimit dieser Entwicklung überschreiten. Gleiches gilt für

¹⁷⁹ Quelle: Eigene Darstellung.

¹⁸⁰ <http://thmanager.sourceforge.net/>, letzter Zugriff am 16.07.2009.

die Implementierung einer anderen Datenbank, welches zu Problemen bei der Erstellung eines Empfehlungssystems führen kann. Denn dies soll in die bestehende Umsetzung integriert werden.

4.3 Vokabularmanager

4.3.1 Anforderungsdefinitionen

Der Vokabularmanager soll im Wesentlichen dazu dienen, die Relationen zwischen den Tags einzugeben. Da eine RDF Ontologie einen allgemeinen Graphen und nicht in einem Baum beschreibt, wird ein Startknoten benötigt. Auf Grund möglicher Zyklen, soll die Navigation durch den Graphen dadurch erfolgen, dass zu dem aktuellen Element lediglich die Ober- und Unterklasse sowie die ähnlichen Elemente angezeigt werden. Durch das Anklicken eines Elementes in der Liste wird dieses ausgewählt. Es werden dann zu diesem ausgewählten Element die verschiedenen Zugehörigkeiten wie die Oberklassen, Unterklassen und ähnlichen Elemente aufgelistet. Nachdem in der Navigation ein Tag ausgewählt wurde, kann diesem ein neuer Tag zugeordnet werden. Dabei gibt es die Möglichkeit das neue Schlagwort als Unterklasse oder ähnlichen Tag zu definieren. Auch soll es die Möglichkeit geben einen Tag mehrfach zuzuordnen, so dass er als Unterklasse mehrerer Schlagworte auftauchen kann.

4.3.2 Umsetzung Vokabularmanager

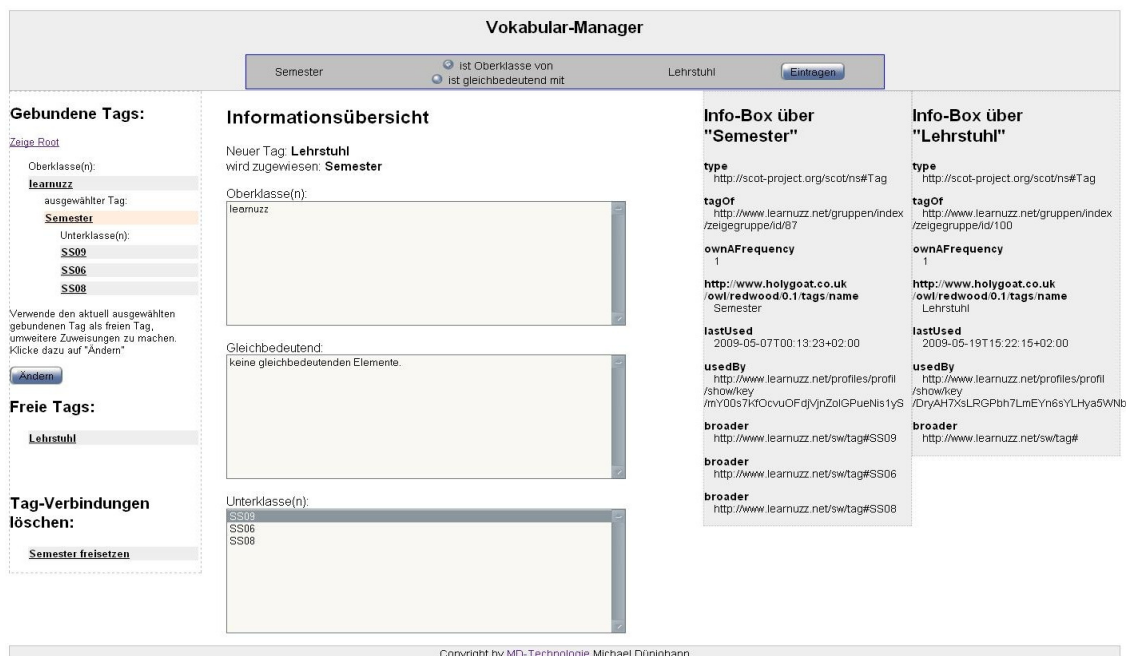


Abb. 16: Der Vokabularmanager¹⁸¹

¹⁸¹ Quelle: Eigene Darstellung

Der Aufbau des Vokabularmanagers erfolgt analog zu den oben beschriebenen Anforderungen. An der linken Seite befinden sich zunächst die gebundenen Tags. Dies sind die dem Graph bereits zugeordneten Tags. In der oberen Ecke gibt es einen Link, um zum Ausgangsknoten zurückzukehren, falls der Benutzer sich tief verschachtelt im Graph befindet. Darunter werden die drei Zuordnungsebenen angezeigt. In der ersten Ebene werden die Oberklassen zum Tag angezeigt, in der zweiten Ebene der Tag an sich, welcher im folgenden Zuweiser genannt wird. Unter dem Zuweiser befinden sich alle Tags, die als ähnlich eingestuft wurden. In der dritten Ebene werden alle Unterklassen des Zuweisers präsentiert. Durch Anklicken eines Tags in dieser Navigation wird dieser als neuer Zuweiser ausgewählt und die Navigation dementsprechend mit den passenden Oberklassen, Unterklassen und ähnlichen Elementen neu aufgebaut. Die Navigation besteht in jedem Fall aus nur drei Ebenen. Durch diesen minimalen Auszug aus dem Graphen wird vermieden, dass das Programm durch Zyklen in einer Endlosschleife landet. Unter der Navigation der Zuweiser-elemente werden die freien Tags, also die, welche noch keine Zuordnung zum Graphen besitzen, angezeigt. Durch Anklicken wird ein neuer Tag ausgewählt. Im oberen Bereich werden nun auf der linken Seite der Zuweiser und auf der rechten Seite der neue Tag angezeigt. Es besteht jetzt die Möglichkeit den Zuweiser als Oberklasse des neuen Tags oder als gleichbedeutend zu definieren. Die Option den neuen Tag als Oberklasse des Zuweisers zu definieren, wird nicht explizit angeboten. Dies soll vermeiden, dass der neue Tag keine Oberklasse hat und folglich nur schwer im Graph zu finden ist. Ein Tag wird automatisch zu einer Oberklasse, sobald er als Zuweiser für einen neuen Tag verwendet wird.

Mit dem Button „Ändern“ wird ein Zuweiser in die Position eines neuen Tags gebracht, so dass er erneut zugewiesen werden kann. Damit wird die Forderung erfüllt, dass ein Tag nicht nur eine Zuordnung haben kann. Außerdem besteht die Möglichkeit Tags wieder freisetzen zu können. Dies ist für jeden Zuweiser möglich, solange er keine Unterklassen besitzt. Diese Restriktion verhindert, dass untergeordnete Elemente keine Verbindung zum restlichen Graphen besitzen. Dazu gibt es noch einige für den Administrator relevante Anzeigen. In der rechten Spalte werden Informationen über den neuen Tag angezeigt und in der linken Spalte werden Informationen über den Zuweiser Tag angezeigt. Dies ermöglicht einen Tag genauer anzusehen und weitere Eigenschaften in Erfahrung zu bringen. In der Mitte des Vokabularmanagers befinden sich Übersichtsboxen in denen redundant zur Navigation jeweils die zum Zuweiser gehörigen Ober- und Unterklassen sowie die ähnlichen Tags gelistet werden. Dies soll bei großen Datenmengen der Übersichtlichkeit dienen.

4.4 Empfehlungssystem

Nach der Entwicklung des Vokabularmanagers und der Auswahl der Werkzeuge sowie der Systemrestriktionen kann nun ein Empfehlungssystem erstellt werden.

4.4.1 Anforderungsdefinitionen

Dieses soll auf dem Portal learnuzz.net implementiert werden und an bekannte Systeme wie web-news.de oder amazon.de, siehe Kapitel 3, angelehnt sein. Auf der learnuzz Plattform gibt es zum jetzigen Zeitpunkt zwei für den Anwender interessante Kategorien, ein Gruppensystem und ein

Fragensystem, in dem die Benutzer verschiedene Gruppen beziehungsweise Fragen anlegen können. Betrachtet werden nun nur die Gruppen. Jede angelegte Gruppe wird mit Tags versehen, welche die Grundvoraussetzung für Empfehlungen bilden. Anhand dieser Tags werden Empfehlungen für weitere Gruppen erstellt, welche für den Benutzer interessant sein könnten. Neben den Empfehlungen für weitere Gruppen, soll es ebenso Empfehlungen für andere hierzu passende Inhalte wie den Fragen geben. Die Empfehlungen einer Kategorie werden in einer Box gebündelt, so dass bei jeder Gruppe verschiedene Empfehlungsboxen für die einzelnen Inhalte angezeigt werden. Diese Empfehlungsboxen existieren auch zusammen mit anderen Inhalten wie den einzelnen Fragen.

4.4.2 Komponenten

Zum Empfehlungssystem gehören neben den bereits erwähnten Anforderungen weitere Komponenten, welche dieses System unterstützen oder für dieses notwendig sind. Eine wichtige unterstützende Komponente ist der Vokabularmanager. Weitere notwendige Komponenten sind die benötigten Klassen zur Abfrage und zur Speicherung der Anfragen des Empfehlungssystems. Außerdem wurden noch einige weitere Komponenten integriert, welche die Arbeit für den Plattformbetreiber erleichtern sollen.

Eine dieser in die Plattform integrierten Komponenten ist eine Übersicht über die verschiedenen Elemente, welche von der SCOT Ontologie gespeichert wurden. Zum einen wurde eine Übersicht für Tags erstellt. Diese listet alle Tags im System auf und ordnet sie nach der Häufigkeit ihres Auftretens. Außerdem werden durch Klick auf den Tag Detailinformationen ausgegeben. Angezeigt werden folgende zum Tag gespeicherte Informationen:

scot:tagOf	Die einzelnen Objekte, also die Gruppen oder ähnliches, auf die der Tag verweist
scot:ownAFrequency	Die Anzahl wie oft der Tag verwendet wurde
tags:name	Der Name des Tags, also seine Beschriftung
rdfs:type	Die Angabe, dass dieses Element vom Typ „Tag“ ist
scot:usedBy	Die Benutzer, welche den Tag bei einem Objekt getaggt haben
scot:lastUsed	Das Datum, wann dieser Tag zuletzt einem Objekt zugeordnet wurde
scot:broader	Die Angabe welche Tags diesem Tag untergeordnet sind
scot:spellingVariant	Zu welchen Tags dieser Tag ähnlich ist

Tabelle 8: Eigenschaften eines Tags in der Ontologie

Die letzten beiden Angaben müssen nicht zwingend vorhanden sein, da diese erst durch den Vokabularmanager gesetzt werden.

Zum anderen wird eine Übersicht über alle im System vorhandenen Tag Clouds gegeben. Diese entsprechen den jeweiligen Kategorien wie den Gruppen. Auch zu den einzelnen Wolken lassen sich detaillierte Informationen anzeigen. Folgende Informationen sind abrufbar:

rdfs:type	Diese Angabe definiert dieses Element als Tag Cloud
scot:contains	Dies listet die Tags auf, die zu dieser Tag Cloud gehören
scot:totalItems	Dies ist die Anzahl der Objekte, die dieser Tag Cloud zugeordnet wurden, zum Beispiel die Anzahl der Gruppen
scot:totalTags	Diese Zahl gibt die Anzahl der Tags an, die zu dieser Tag Cloud gehören
scot:tagSpace	Diese Angabe definiert, auf welchem Raum diese Tag Cloud gültig ist, hier „learnuzz.net“
scot:taggingActivity	Jede Taggingaktivität wird der entsprechenden Tag Cloud zugeordnet und kann für Auswertungen abgerufen werden

Tabelle 9: Eigenschaften einer Tag Cloud in der Ontologie

Als drittes können die ermittelten Zusammenhänge angezeigt werden. Das sind in unserem Fall Paare von Tags, welche beim Abspeichern eines Schlagwortes erstellt wurden. In der Liste werden die Zusammenhänge nach der Häufigkeit ihres Auftretens sortiert. So kann schnell erkannt werden, welche Zusammenhänge existieren und welche besonders stark sind. Als letzte Funktion in dieser Übersicht gibt es die Möglichkeit sich alle getätigten Aktivitäten anzuschauen. Diese werden sortiert nach der Zugehörigkeit ihrer Tag Cloud ausgegeben.

Neben den Übersichtsfunktionen gibt es zusätzlich die Möglichkeit manuell Tag Clouds und Tags nachzupflegen. Für die Tag Clouds ist dies erforderlich, wenn eine neue Kategorie angelegt werden soll. Dabei muss lediglich der Tag Cloud Name eingegeben werden. Die Möglichkeit Tags gezielt einzugeben eröffnet dem Experten des Vokabularmanagers die Chance noch nicht existente Tags anzulegen, welche ihm eine bessere Strukturierung ermöglichen. So können Oberklassen angelegt werden um dazugehörige Tags unterzuordnen. Die Eingabe eines neuen Tags verlangt die Angabe des Tagnamens, der zugehörigen Gruppe, die Angabe eines Objektes, dem der Tag zugeordnet wird sowie einen Benutzer, der den Tag angelegt hat.

Eine schon implementierte, aber noch nicht genutzte Funktion ist, dass sich der Benutzer einen Überblick über die Tags verschaffen kann, welche er vergeben hat. Diese können in einer Tag Cloud dargestellt werden, welche die Interessen des Benutzers widerspiegelt. Durch Anklicken eines einzelnen Tags werden dem Benutzer die zu diesem Tag gehörenden Objekte angezeigt. So hat er die Möglichkeit schnell weitere zu ihm passende Objekte zu finden.

4.4.3 Funktionsbeschreibungen

Diese bisherigen Funktionen bauen auf einem Konstrukt auf, dass es ihnen erlaubt Daten im RDF Format in der Datenbank zu speichern. Dazu wird die Bibliothek ARC2 verwendet. Diese bietet allerdings nur die grundsätzliche Möglichkeit einer Schnittstelle, um Anfragen an die Datenbank zu bearbeiten. Was fehlt sind die expliziten Funktionen, welche die SCOT Ontologie nutzen und auf die Bedürfnisse des zu entwickelnden Systems angepasst werden müssen. Diese Funktionen werden in zwei Klassen gegliedert.

Funktionsbeschreibung ScotStore

Die Klasse ScotStore ist für das Speichern der Informationen, anhand der SCOT Ontologie, erstellt worden. Dazu werden Anfragen mittels der Anfragesprache SPARQL+ erstellt, welche über die ARC2 Funktionalität in der Datenbank gespeichert werden. SPARQL+ ist eine Erweiterung von SPARQL und enthält minimale Veränderungen.

Die erste Funktion, welche auch ausgeführt werden muss, wenn das System neu angelegt wird ist „storeTagCloud“. Sie dient zum Speichern einer neuen Tag Cloud oder zum Ergänzen einer bestehenden. Als Parameter benötigt diese Funktion eine Kategorie, welche in diesem System eine Gruppe oder eine Frage wäre. Außerdem wird eine Sammlung mit Tags erwartet, welcher der Wolke hinzugefügt werden können. Diese Tags gehören zu einem Objekt, also einer Gruppe oder Frage, welches ebenfalls übergeben wird. Sollte die Tag Cloud noch nicht existieren, wird sie neu angelegt, wobei in diesem Fall die Parameter für die Tags und für das Objekt leer sein können. Ansonsten wird die bestehende Tag Cloud um die neuen Tags und das zugehörige Objekt erweitert und die Anzahl, der in der Tag Cloud befindlichen Tags und Objekte, erhöht.

Die Funktion „storeFirstTag“ legt den ersten Tag an, der im Graphen für den Vokabularmanager den Ausgangsknoten bildet. Dieser Tag verweist daher auf kein Objekt. Mit „storeTag“ wird ein neuer Tag gespeichert oder ein bestehender um Informationen ergänzt. Als Parameter erhält diese Funktion den Tagnamen, woraus der Taglink generiert wird, die Angabe, zu welcher Gruppe er gehört, auf welches Objekt der Tag verweist und welcher Benutzer ihn angelegt hat. Sollte der Tag schon existieren, wird er um die Angabe des eventuell neuen Objektes ergänzt, der neue Benutzer wird hinzugefügt, das Datum der letzten Benutzung aktualisiert und die Anzahl des Auftretens des Tags wird um einen Zähler erhöht. Sollte der Tag noch nicht existieren, werden diese Werte neu angelegt und die Zuweisung wird durch „skos:broader“ auf einen Link gesetzt, welcher mit „tag#“ endet. Dies zeigt dem System, dass der Tag noch keine konkrete Zuordnung besitzt. Die Beziehung zu ähnlichen Klassen werden durch „scot:spellingVariant“ bei Bedarf später ergänzt.

Die Funktionen „storeTagOberklasse“, „storeTagUnterklasse“ und „storeTagAehnlich“ arbeiten sehr analog zueinander und werden nur vom Vokabularmanager benutzt. Sie bekommen jeweils als Parameter den Tagnamen des Tags, welcher als Ober-, Unter-, oder Ähnlicheklasse definiert werden soll und als zweites den Namen des Tags, welcher als zuweisendes Element gilt. Bei einem neuen Tag, welcher noch keine Zuweisung besitzt, wird der leere Pfad von „skos:broader“ gelöscht und die neue Abhängigkeit eingefügt. Ansonsten wird lediglich letzteres durchgeführt. Bei den ähnlichen Tags wird außer der Ähnlichkeit für den neuen Tag auch redundant die Ähnlichkeit im

zuweisenden Tag gespeichert. Außerdem erbt der neue Tag alle Oberklassen des zuweisenden Tags.

Die Funktion „tagVerbindungLoeschen“ löscht alle Beziehungen des übergebenen Tags zu anderen Tags. Es wird eine Überprüfung gefahren, ob der Tag Unterklassen besitzt. Dies darf nicht der Fall sein, da ansonsten einige Tags ihre Verbindung zum Graphen verlieren.

Mit „storeTagging“ wird die Tagging Aktivität gespeichert. Als Parameter wird eine Sammlung von Tags übergeben, die Kategorie, das getaggte Objekt und der User, der die Aktivität tätigte. Zusätzlich wird das Datum gespeichert, an dem diese Aktion durchgeführt wurde.

Als letztes werden die Zusammenhänge mit der Funktion „storeCooccurrence“ gespeichert. Dabei werden beim Anlegen eines Objektes alle dazugehörigen Tags übergeben. Diese Tags werden alphabetisch sortiert und dann paarweise gespeichert. Durch die Sortierung wird erreicht, dass Elemente nicht doppelt abgespeichert werden müssen. Es wird also nicht vorkommen, dass die Tags „BWL“ und „Klausur“ sowohl zu „BWLKlausur“ und zu „KlausurBWL“ zusammengefügt werden. Dies verhindert eine redundante Speicherung. Sollte ein Zusammenhang schon bestehen, wird die Anzahl seines Auftretens um einen Zähler erhöht. Andernfalls wird ein neues Zusammenhangspaar angelegt.

Funktionsbeschreibung ScotQuery

Das wesentliche Einsatzgebiet von SPARQL ist nicht das Speichern von Daten, sondern deren geschickte Abfrage. Die Abfragen für dieses System wurden in der Klasse ScotQuery gebündelt und decken folgende Funktionen ab:

Mit der Funktion „getFirst“ wird der für den Graphen als Ursprungselement definierte Tag abgefragt. Dieser findet bei dem Vokabularmanager Verwendung, bei dem ein Startpunkt zum navigieren durch den Graph benötigt wird.

Als weitere wichtige Funktionen für den Vokabularmanager sind „getOberklassen“, „getUnterklassen“ und „getAehnliche“ zu erwähnen. Diese bekommen als Parameter einen Tag übergeben und suchen zu diesem jeweils die Ober-, Unter- oder ähnlichen Klassen heraus. Zusätzlich wird die Anzahl des Auftretens der jeweiligen Tags in der Klasse zurück gegeben und die zurückgegebenen Schlagwörter nach dieser absteigend sortiert. Jede dieser drei Funktionen hat noch zwei ähnliche Aufrufe, die jeweils eine Besonderheit bieten. Die erste Variante, welche als Funktionsnamen die Ergänzung „Gmax“ besitzt, gibt die ersten fünf Ergebnisse zurück. Da das Ergebnis nach der Anzahl sortiert ist, werden die fünf passenden Tags, welche am häufigsten Auftreten, zurückgegeben. Analog dazu gibt es drei Funktionen mit der Ergänzung „Gmin“. Diese geben ebenfalls nur die ersten fünf Ergebnisse zurück, allerdings ist die Rückgabeliste dieses mal aufsteigend, nach der Anzahl, sortiert. Dies führt dazu, dass die fünf am wenigsten genutzten Tags zurückgegeben werden. Der genaue Einsatz dieser Varianten wird später in der Beschreibung des Empfehlungsalgorithmus deutlich. Durch die Funktion „getFreieTags“ werden für den Vokabularmanager alle noch nicht zugeordneten Tags ausgegeben.

Für die anfangs erwähnten Übersichtskomponenten befinden sich in dieser Klasse die nötigen Funktionen. Mit „getTagInfo“ und einem Tag als Parameter werden alle zum Tag gehörenden Informationen abgefragt. Mit dem Parameter der Tag Cloud fragt die Funktion „getTagcloudInfo“ analog die Daten einer Tag Cloud ab, ebenso wie die Funktion „getTaggingActivitiesInfo“ die Details einer Tag Aktivität ermittelt. Ebenfalls für die Übersichtskomponenten wurden die Funktionen „getTags“ zur Abfrage aller Tags, „getTagClouds“ zur Beschaffung aller vorhandenen Tag Clouds und „getCooccurrence“ zur Anzeige der Zusammenhänge erstellt. Mit „getTaggingActivities“ werden alle Aktivitäten gruppiert und nach der Zugehörigkeit ihrer Tag Cloud ausgegeben.

Für den Empfehlungsalgorithmus müssen für die Ausgabe nicht nur die Tags vorhanden sein, sondern auch die dazugehörigen Objekte ausgegeben werden. Die Funktion „getTagItems“ fragt alle zu dem mittels Parameter übergebenen Tag gehörigen Objekte ab und gibt sie zurück. Ebenso verfährt die Funktion „getCooccurrenceItems“, welche die zum Zusammenhangsparameter gehörigen Objekte zurück liefert. Die Funktion „getUserTags“ beschafft alle Tags, die ein bestimmter Benutzer im Laufe der Zeit eingegeben hat. Dazu wird lediglich der Link zum Benutzerprofil übergeben, anhand dessen die Tags ermittelt werden können.

4.4.4 Empfehlungsalgorithmus

Als letzte Funktion in der Klasse ScotQuery ist der Empfehlungsalgorithmus vorhanden.

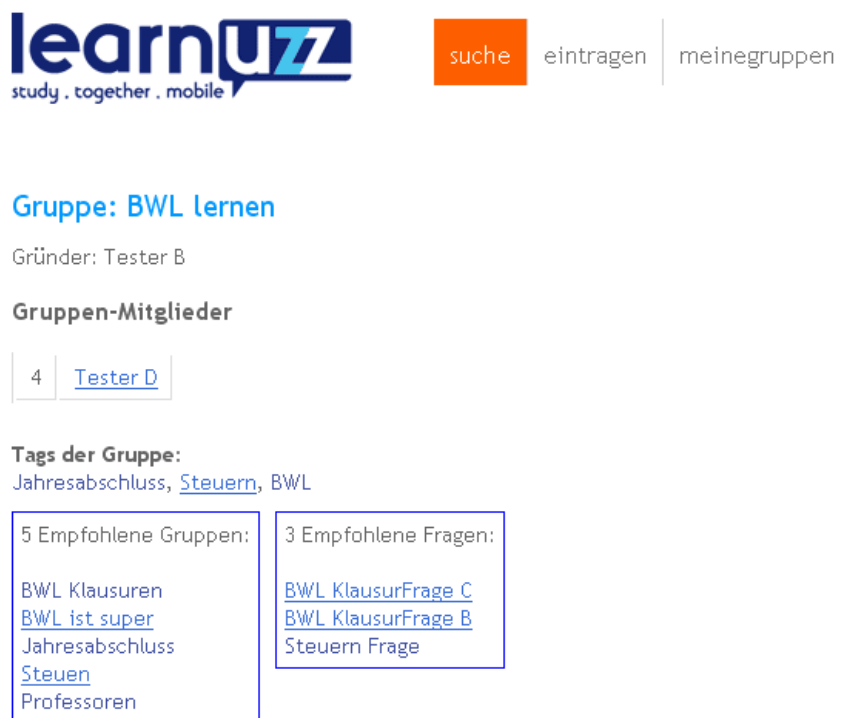


Abb. 17: Ausgabe von Empfehlungen zu einer Gruppe¹⁸²

¹⁸² Quelle: Eigene Darstellung.

Algorithmus

Der Algorithmus besteht aus fünf verschiedenen Teilen, welche alle von den Tags des jeweiligen Objektes abhängig sind. Die verschiedenen Teile bestehen aus den Zusammenhangs-Tags, den Tags welche zum Objekte gehören und den passenden Ober-, Unter- und ähnlichen Klassen.

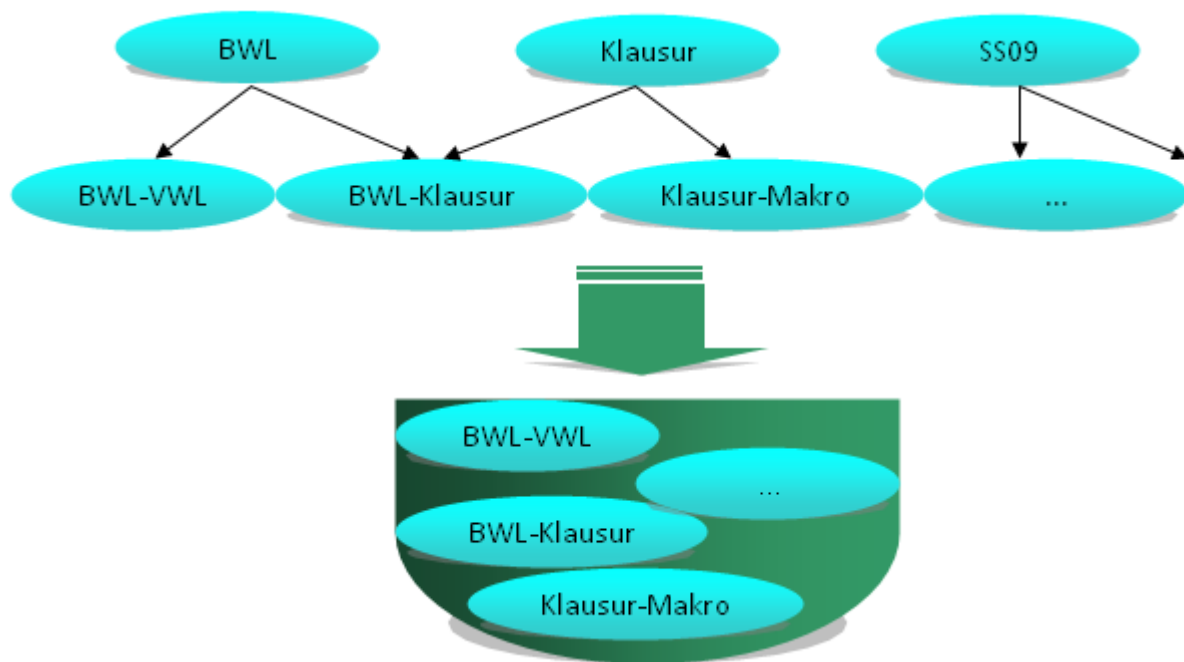
Als Übergabeparameter gibt es zum einen die Kategorie, also ob es sich um eine Gruppe oder eine Frage handelt, sowie das Objekt, beispielsweise die konkrete Gruppe oder Frage. Anhand des Objektes werden zuerst die dazugehörigen Tags ermittelt. Beispielsweise handelt es sich bei dem Objekt um eine Frage aus einer BWL Klausur. Dieser Frage sind die drei Tags „BWL“, „Klausur“ und „Sommersemester 2009“ zugeordnet. Diese werden, bildlich gesprochen, in einem Topf gesammelt.



Abb. 18: Objekte eines Tags sammeln¹⁸³

Zu jedem Tag aus dieser Sammlung werden nun die verschiedenen Abhängigkeiten untersucht. Zum einen werden zu jedem Tag die Zusammenhänge ermittelt und diese ebenfalls in einem Topf gesammelt. Als ein Beispiel für einen Zusammenhang wäre „BWL“ und „Klausur“. Allerdings kann der Tag BWL auch mit weiteren vom Objekt unabhängigen Tags wie „VWL“ verbunden sein.

¹⁸³ Quelle: Eigene Darstellung.

Abb. 19: Sammlung der Zusammenhang-Tags¹⁸⁴

Im nächsten Schritt werden die Ober-, Unter- und ähnlichen Klassen ermittelt. Der Tag „Vorlesung“ könnte dabei eine Oberklasse sein und „Steuern“ eine Unterklasse von „BWL“ bilden. Als ähnlich könnte der Tag „Betriebswirtschaftslehre“ angesehen werden. Es gibt hierbei eine Unterteilung in die minimalen Oberklassen und die maximalen Oberklassen, wenn diese Klasse näher betrachtet wird. Dies bedeutet, zu allen zum Objekt gehörenden Tags werden diese beiden Arten von Oberklassen herausgesucht. Bei dem vorgegebenen Beispiel könnte neben der Klasse „Vorlesung“ ebenfalls die Oberklasse „Wirtschaftswissenschaften“ existieren. Ersteres könnte eine Häufigkeit von „25“ besitzen, was bedeutet, dass der Tag „Vorlesung“ „25“ mal verwendet wird. Letzteres könnte die Häufigkeit „2“ besitzen. Für die maximalen Oberklassen werden für jeden Tag des Objektes die Tags zurückgegeben, welche als Oberklasse fungieren und am häufigsten vorkommen, also der Tag „Vorlesung“. Wobei der Algorithmus für jeden Tag des Objektes fünf Oberklassen Tags zurück gibt, statt wie in unserem Beispiel nur eine. Häufig verwendete Tags können auf der einen Seite den Nachteil haben, dass sie relativ allgemeine Tags darstellen und ungenaue Ergebnisse liefern. Andererseits werden sie häufig eingesetzt und sind dadurch mit den für die Benutzer interessanten Objekten verbunden. Natürlich können hierbei die Tags des Objektes gleiche, häufig genutzte Oberklassen zurückgeben. Denn der Tag „Klausur“ kann wie der Tag „BWL“ ebenfalls als Oberklasse den Tag „Vorlesung“ besitzen. Da jedoch die Oberklassen in einen Topf landen, sind diese Überschneidungen vorerst nicht relevant. Analog zu den maximalen Oberklassen gibt es auch die minimalen Oberklassen. Dies sind für jeden Tag des Objektes fünf Oberklassen, welche am wenigsten genutzt werden, wie zum Beispiel der Tag „Wirtschaftswissenschaften“. Da bei den maximalen Oberklassen das Problem festgestellt wurde, dass häufig genutzte Tags zu allgemein sein können, bieten die wenig genutzte Tags die Chance sehr spezifisch zu sein und damit bessere

¹⁸⁴ Quelle: Eigene Darstellung.

Ergebnisse zu liefern. Alle minimalen Oberklassen landen ebenfalls in einen Topf. Die Vorgehensweise mit den maximalen und minimalen Tags wird gleichermaßen für die Unterklassen und für die ähnlichen Klassen angewandt.

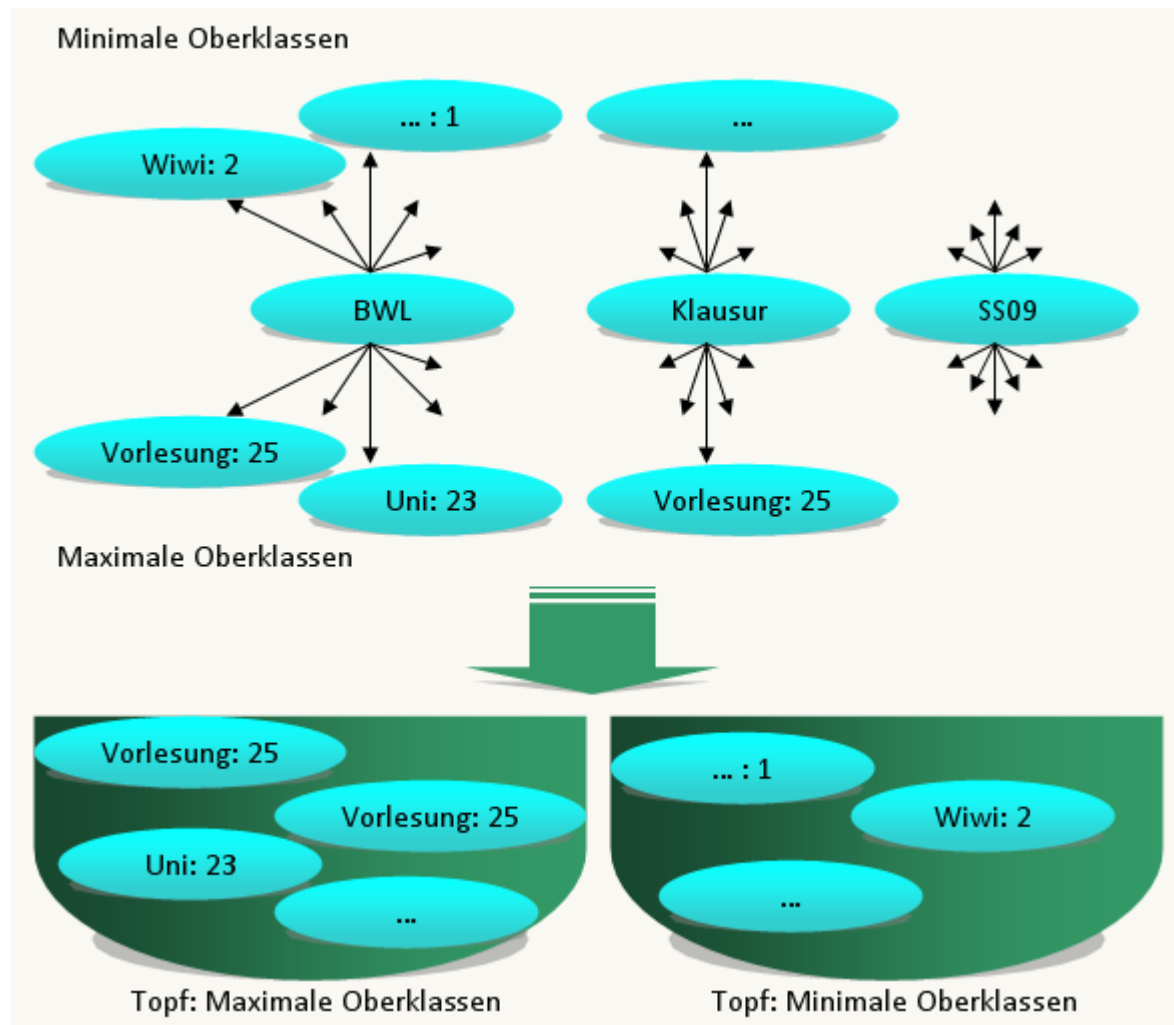


Abb. 20: Minimale und maximale Oberklassen gesammelt¹⁸⁵

Das Ergebnis sind acht Töpfe. Zum einen die Tags des Objektes wie „BWL“, „Klausur“ und „Sommersemester 2009“, die Zusammenhänge, die maximalen Oberklassen, die minimalen Oberklassen, die maximalen Unterklassen, die minimalen Unterklassen, die maximalen ähnlichen Klassen und die minimalen ähnlichen Klassen. In den einzelnen Töpfen werden die Tags zum Schluss nach ihrer Häufigkeit sortiert, so dass die wichtigsten oben liegen.

Nachdem alle Tags ermittelt worden sind, welche für das Objekt, also der Frage bezüglich einer BWL Klausur, relevant sein können, und in verschiedene Töpfe unterteilt worden sind, werden zu jeden Topf die mit den Tags verbundenen Objekte aus der Datenbank geholt. Das heißt, für jeden Tag aus einem Topf werden die Objekte gesucht, welche zu diesem Tag und auch zu der jeweiligen Kategorie, in unserem Fall Fragen, gehören. Allerdings kann nicht nur ein Tag in einem Topf

¹⁸⁵ Quelle: Eigene Darstellung.

mehrfach vorhanden sein, auch ein Objekt kann wiederholt vorkommen. Dies wird einerseits dadurch verursacht, dass ein Tag in einem Topf mehrfach enthalten ist, andererseits auf Grund dessen, dass verschiedene Tags auf das gleiche Objekt verweisen. Beides ist explizit gewollt. Zu den einzelnen Töpfen mit Tags sind nun analog Töpfe mit Objekten erstellt worden. Als nächstes wird die Häufigkeit mit der ein Objekt in einem Topf vorkommt gezählt. Danach werden die gezählten Objekte nach der Häufigkeit ihres Auftretens sortiert, so dass diejenigen, welche am häufigsten vorkommen, im Topf oben liegen. Durch das Zählen der Häufigkeit wurde die Schnittmenge der Objekte ermittelt, die am stärksten mit dem aktuellen Objekt in Verbindung stehen. Im vorherigen Schritt wurde dies schon dadurch forciert, dass Tags in einer Klasse mehrfach in einem Topf vorkommen können, wenn es mehrere Verbindungen zu ihnen gibt. Als Ergebnis entstehen wieder acht Töpfe mit Objekten, in welchen die Inhalte nach ihrer Häufigkeit sortiert sind.

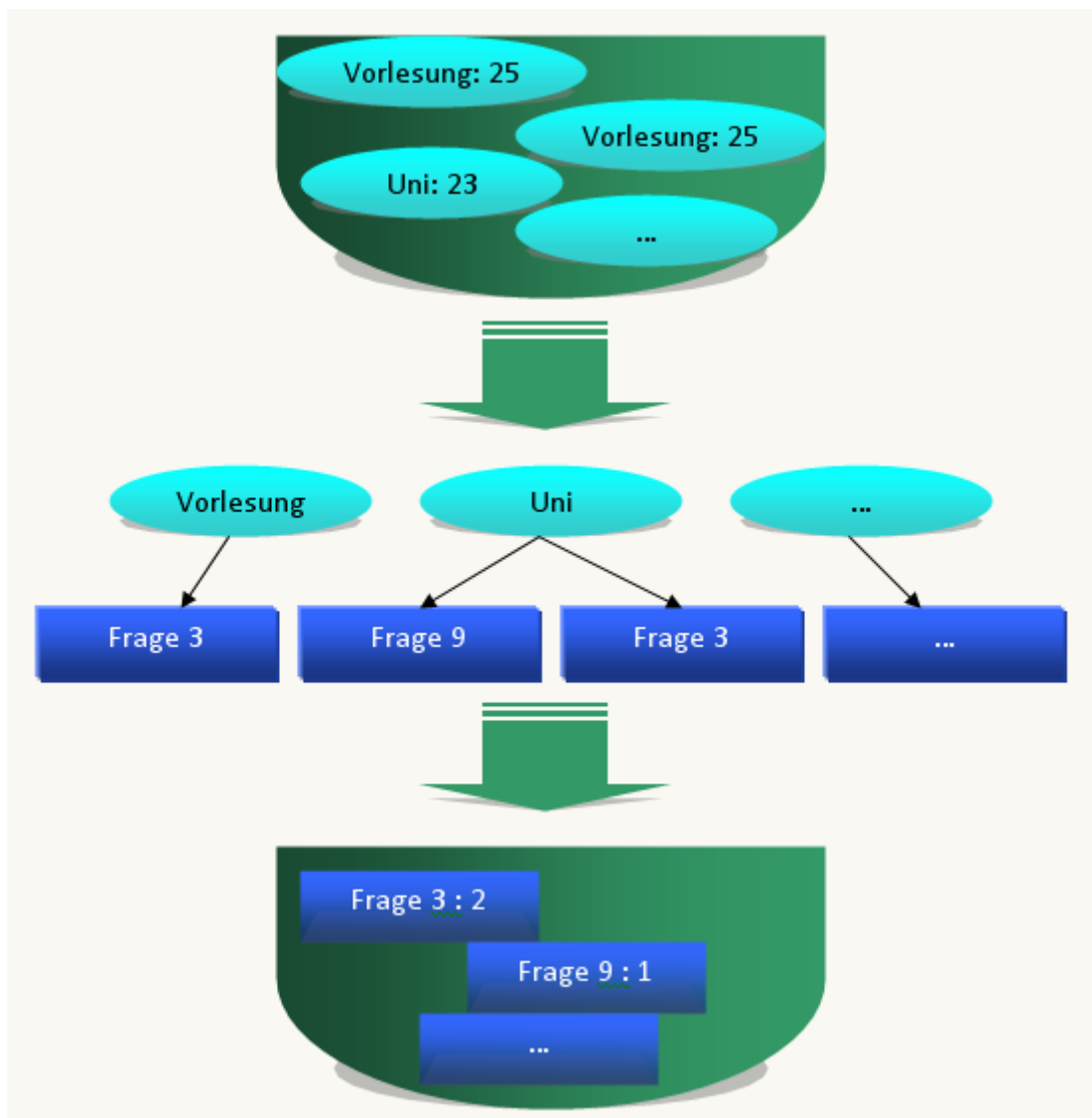


Abb. 21: Ermitteln der zum Tag gehörenden Objekte¹⁸⁶

¹⁸⁶ Quelle: Eigene Darstellung.

Im nun letzten Schritt wird aus den verschiedenen Töpfen die Ausgabe erstellt. In dieser sollen die wahrscheinlichsten Treffer oben stehen. Da die paarweise auftretenden Zusammenhangs-Tags die stärkste Bindung zwischen Schlagwörtern darstellen dürften, wird ihnen besondere Aufmerksamkeit geschenkt. In der Ausgabe werden die beiden häufigsten Objekte aus dem Zusammenhang Topf an oberster Stelle gesetzt. Dabei wird gefiltert, so dass das ausgegebene Objekt weder das aktuelle Objekt ist, noch das das Objekt des Ursprungsknotens im Graph. An dritter Stelle in der Ausgabe erscheint das oberste Objekt aus dem Topf mit den Tags „BWL“, „Klausur“ und „Sommersemester 2009“, also die Tags, die direkt zum aktuellen Objekt gehören. Da das Objekt mit diesen Tags gekennzeichnet wurde, sollte davon ausgegangen werden, dass andere Objekte, auf die diese Tags zeigen ebenfalls von starkem Interesse für den aktuellen Benutzer sein könnten. Dabei wird das oberste Objekt aus diesem Topf nur genommen, wenn es noch nicht der Ausgabe zugeordnet wurde. Andernfalls wird der Topf der Objekte solange in Reihenfolge der Häufigkeit durchsucht, bis ein Objekt gefunden wurde, welches neu ist. Sollten sich bereits alle Objekte in der Liste befinden, wird kein Objekt hinzugefügt. An vierter Stelle der Ausgabe steht das Objekt, welches sich zuoberst im Topf der maximalen Ähnlichkeit befindet. An fünfter Stelle steht das Objekt aus dem Topf der minimalen Ähnlichkeit. Auch hier gilt die Restriktion, dass sich das Objekt noch nicht in der Ausgabe befinden darf oder dass es sich nicht um das aktuelle Objekt beziehungsweise um das Objekt des Ursprungsknotens handeln darf. Da ähnliche Tags sehr eng verbunden sind mit den Tags des aktuellen Objektes, folgen sie in der Ausgabeordnung direkt hinter den Vorschlägen zu den Tags des aktuellen Objektes. An der nächsten Stelle, also an Ausgabestelle sechs und sieben erscheinen die häufigsten Objekte aus den Töpfen der maximalen Unterklasse und danach der minimalen Unterklasse. Diese beiden Töpfe werden vor den Oberklassen ausgegeben, da Unterklassen eine Spezialisierung der Tags darstellen und zum aktuellen Objekt spezielle Unterpunkte abdecken könnten. Aus den beiden letzten Töpfen der maximalen Oberklasse und der minimalen Oberklasse werden die letzten beiden Ausgabeobjekte herausgezogen. Auch hier wird ebenfalls wieder das noch nicht verwendete Objekt mit der größten Häufigkeit genommen. Oberklassen decken den Bereich der Verallgemeinerung ab und können so ungenau getaggte Objekte finden. Insgesamt wird durch die Anordnung eine Reihenfolge in der Ausgabe erzeugt, in der die besten Empfehlungen als erstes erwähnt werden sollten.

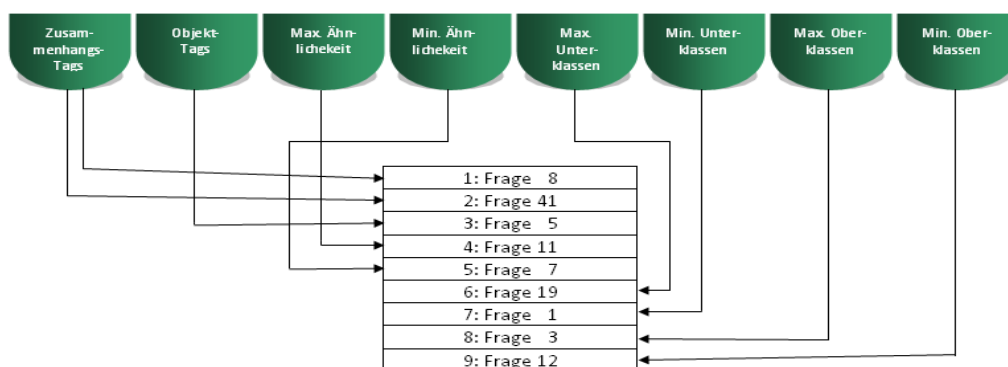


Abb. 22: Objekte in Ausgabeliste einfügen¹⁸⁷

¹⁸⁷ Quelle: Eigene Darstellung.

Vorteile

Die Ausgabe besteht letztendlich aus einer Mischung von verschiedenen Auslegungen der Tags die zum aktuellen Objekt gehören. Dabei gibt es grundsätzlich fünf verschiedene Ansätze, welche zum Teil in maximale und minimale Kriterien unterteilt worden sind. Diese Mischung soll sicherstellen, dass nicht passende Empfehlungen in den einzelnen Ansätzen zwar nicht grundsätzlich ausgeschlossen werden, aber durch diesen Mix keine große Gewichtung im Gesamtergebnis der Ausgabe bekommen.

Nachteile

Wenn es nur wenige Ergebnisse im niedrigen einstelligen Bereich in den einzelnen Töpfen der Tags gibt, kann das Gesamtergebnis verfälscht werden. Bei der Annahme, dass der Topf mit den Tags, welche zum aktuellen Objekt gehören, gute Ergebnisse liefert, die anderen Töpfe aber nahe zu leer sind, ist es möglich, dass die Ausgabemenge sehr gering ist, obwohl im ersten Topf mehrere gute Ergebnisse vorhanden sind. Das liegt daran, dass aus jedem Topf nur ein Objekt, beziehungsweise für den Zusammenhangstopf zwei, ausgewählt werden. Wenn diese Auswahl die Ergebnisse der anderen Töpfe wieder spiegelt, werden dort auf Grund der Vermeidung von doppelten Ausgaben keine Objekte herausgenommen. Ebenfalls möglich ist eine negative Wirkung der Mischung im Topf, wenn zwar alle Töpfe gut gefüllt sind, aber nur in einem gute Empfehlungen vorhanden sind. Denn auch in diesem Fall wird aus jedem Topf nur die begrenzte Anzahl an Objekten gezogen. Die Ausgabemenge ist in diesem Fall umfangreich, enthält aber kaum brauchbare Ergebnisse.

Eine weitere Einschränkung kommt aus dem technischen Bereich. Die benutzte Bibliothek ARC2 ist nur für das englische Alphabet ausgelegt und hat Probleme mit deutschen Umlauten, sofern diese nicht in Hexadezimal umgerechnet wurden. Auch Leerzeichen in den Tags sind nicht erlaubt. Falls dies doch geschieht, werden die Tags nach dem Leerzeichen abgeschnitten. Generell sind nur Zeichen erlaubt, die auch für eine URL gültig sind. Systeme wie das eingesetzte Zend Framework wandeln die nicht erlaubten Zeichen automatisch in für die Bibliothek gültige Hexadezimaldarstellung um.

Diese beiden genannten und bekannten Nachteile konnten in einer künstlich erstellten Testumgebung konstruiert werden. Da die Menge an Objekten und Tags auf einer aktiven Plattform jedoch wesentlich größer ist, sollten diese Probleme in der großen Anzahl der Daten untergehen. Dabei wird das gleiche Prinzip angenommen, welches auch für Folksonomies gilt. In der Masse sind Ausreißer nicht relevant. Die Menge an verschiedenen Daten liefert gute Werte, und damit gute Empfehlungen.¹⁸⁸

¹⁸⁸ Vgl. Gaiser (2008), S. 180.

4.4.5 Vergleich zu anderen Empfehlungssystemen

Die Idee eines Empfehlungsalgorithmus ist nicht neu, so dass schon verschiedene Ansätze in dieser Richtung existieren. Insbesondere im E-commerce wurde diese Entwicklung forciert, da durch das Empfehlen von interessanten Produkten dem Kunden nicht nur ein neuer Service geboten wird, sondern auch die Absätze gesteigert werden können. Ein zielgerichtetes Präsentieren von interessanten Artikeln für den Benutzer kann wesentlich effektiver sein als eine unpräzise Platzierung von Werbung auf einer Webseite. Es existieren mehrere Ansätze, um für den Benutzer interessante Objekte durch einen Algorithmus zu finden. Diese Verfahren unterscheiden sich im Wesentlichen durch die Qualität der Ausgabeobjekte und durch ihre Laufzeit. Letzteres ist insbesondere bei großen Datenmengen von Produkten und Kunden ein relevanter Faktor.

Kollaboratives Filtern

Eine Möglichkeit Ähnlichkeiten zwischen den Anwendern zu finden besteht darin, den Kunden mit den anderen in der Datenbank befindlichen Nutzern zu vergleichen. Dadurch können Konsumenten, welche ein ähnliches Kaufverhalten aufweisen miteinander verglichen werden. Kauft beispielsweise Kunde A die Produkte P1, P2, P3 und Kunde B die Produkte P2, P3, P4, kann dem Kunden A das Produkt P4 empfohlen werden. Dies soll eine etwas vereinfachte Darstellung sein und zur Veranschaulichung dienen, da der Algorithmus in Wirklichkeit die Ähnlichkeit zwischen Benutzern dadurch ermittelt, dass jeder Benutzer als n -dimensionaler Vektor dargestellt wird und die Abstände zwischen diesen berechnet werden. Der Nachteil dieser Umsetzung ist, dass der Vergleich rechen- und damit zeitaufwändig ist, insbesondere bei großen Datenbeständen. Denn jeder Benutzer wird mit jedem anderen Kunden verglichen. Wird dabei zur Optimierung jeweils nur eine Stichprobe verglichen, sinkt die Qualität der Ergebnisse.

Cluster Models

Um die Rechenzeit zu optimieren, besteht die Möglichkeit die Kunden zu gruppieren. Ähnliche Kunden werden in einem Cluster gebündelt, so dass bei der Empfehlungssuche nur diese Sammlungen durchsucht werden müssen. Die Übereinstimmung wird wie beim kollaborativen Filtern ebenfalls anhand der gekauften Produkte ermittelt. Bei diesem Algorithmus ist von Vorteil, dass die Berechnung der Cluster nicht am Live-System durchgeführt werden und folglich das laufende System nicht belastet. Der Nachteil liegt in der Ungenauigkeit, denn Benutzer werden zu Gruppen zusammengefasst und mit den Gruppenmitgliedern in Bezug gesetzt. Damit vergleicht das System nicht zwangsweise die Kunden, welche die größte Ähnlichkeit aufweisen. Denn dies hängt stark davon ab wie gut die Benutzer in die einzelnen Cluster eingeteilt wurden. Eine feinere Einteilung ist zeitlich aufwändiger und nähert sich damit dem kollaborativen Filtern an.

Suchbasierte Methoden

Die bisherigen Algorithmen basierten auf dem Vergleich von Kunden, jedoch nicht auf der Übereinstimmung von Artikeln. Bei der suchbasierten Methode werden die verwandten Produkte herausgesucht. Dies kann anhand verschiedener Kriterien geschehen wie beispielsweise Schlagwörtern, Titeln, Autoren oder Popularität. Bei Benutzern mit nur wenigen gekauften Produkten oder abgegebenen Bewertungen stellt sich das finden ähnlicher Produkte einfacher dar und liefert daher gute Ergebnisse. Bei größeren Datenmengen wird dies schwieriger, da die Gesamtmenge der Informationen zu uneinheitlich sein kann, als dass gute Verwandtschaften gefunden werden. Es muss demzufolge eine Untermenge oder eine Zusammenfassung der Daten verwendet werden, welche zu einer ungenaueren Ergebnismenge führt.

Kollaboratives Produkt-zu-Produkt filtern

Auch der Algorithmus von amazon.com legt den Fokus auf das Produkt. Das Ziel besteht darin, dass die Ähnlichkeiten zwischen den Produkten ermittelt werden. Dazu wird eine Matrix angelegt, in der die Artikel paarweise miteinander verglichen werden. Daraus entsteht eine Tabelle, in der die Produkte enthalten sind, welche auf Grund des Kaufverhaltens der Benutzer miteinander korrelieren. Der Vorteil bei diesem Verfahren ist, dass die Generierung der Matrix nicht am Live-System erfolgen muss und mit geringem Zeitaufwand beliebte Artikel mit einer starken Korrelation ausgelesen werden können.¹⁸⁹

Fazit

Im Vergleich zu anderen Algorithmen wird ersichtlich, dass das in dieser Arbeit entworfene Empfehlungssystem eine Mischung aus bereits vorhandenen Methoden ist. Der Schwerpunkt liegt ebenfalls auf dem Vergleich von Objekten und nicht auf der Übereinstimmung von Benutzern. Ähnlich der Matrix von amazon.com werden hier Zusammenhangs-Tags benutzt, um paarweise Verbindungen zu erkennen und auszuwerten. Auch werden anhand von Schlagwörtern gleichartige Objekte gesucht, wie es ebenso in der suchbasierten Methode geschieht. Zusätzlich findet, mit der Einteilung der Tags in verschiedene Töpfe, ein einfaches Gruppieren statt.

4.4.6 Ausblick

Der entworfene Empfehlungsalgorithmus stellt eine gute Grundlage dar, bietet aber noch Potential für Erweiterungen und Verbesserungen. In erster Linie muss dabei festhalten werden, dass eine ausführliche Auswertung über die Effektivität, in Form einer empirischen Studie, fehlt. Die Ansätze mögen logisch aufgebaut sein, ob sie den Erwartungen des Benutzers entsprechen will noch erforscht werden, woraus Möglichkeiten für eine Erweiterung erkennbar sind. Die von einem Algorithmus empfohlenen Ergebnisse werden, auf Grund ihrer automatischen Generierung, wahrschein-

¹⁸⁹ Vgl. Linden /Smith /York (2003).

lich nie vollkommen zufriedenstellend sein, was jedoch nur von einem Menschen bewertet werden kann. Daher könnte, ähnlich wie bei der Google Suche¹⁹⁰, der Benutzer die Ergebnisse als gut oder nicht zutreffend markieren, so dass dies berücksichtigt werden kann. Auch können noch weitere Faktoren für die Empfehlung eine Rolle spielen. Zu den Fragen könnte es eine generelle Bewertung ihrer Qualität geben die ebenfalls Auswirkungen darauf hat, ob dieses Objekt empfohlen werden sollte. Die Benutzer könnten bei jeder Frage festlegen, ob sie hilfreich war oder nicht. Ebenfalls ist es möglich jede Antwort auf eine Frage zu bewerten, was zu einer zu berücksichtigenden Gesamtbewertung führt. Zugleich könnte auch die Anzahl der Bewertungen ein wichtiges Kriterium sein. Viele Bewertungen drücken aus, dass sich die Benutzer für dieses Objekt interessiert haben.

Ein weiteres großes Forschungsgebiet wird eröffnet, wenn der Vokabularmanager extrapoliert wird. Dieser muss zurzeit manuell von einem Experten gepflegt werden. Es würde erheblichen Aufwand und Kosten sparen, wenn es möglich wäre eine Automatisierung einzuführen. Ansätze hierfür gibt es auf verschiedene Art und Weise. Mittels eines Clustering Verfahrens¹⁹¹ können Zusammenhänge von Tags erkannt und anschließend gruppiert werden. Solche Gruppen können nun einer Obergruppe zugeordnet werden. Dabei ermittelt das Verfahren, wie eng der Zusammenhang zwischen den Tags ist. Angefangen bei Zweiergruppen werden immer mehr, in der Nähe befindliche, Tags hinzugezogen, bis ein Häufungspunkt entsteht. Auf die Obergruppen kann dann ebenfalls wieder ein Clusteringverfahren angewandt werden. Dieses Verfahren ist verwandt mit dem Card Sorting Algorithmus¹⁹². Dabei werden die verschiedenen Karten, in diesem Fall Tags, so sortiert, dass thematisch passende Karten auf einem Stapel landen. Für jeden gebildeten Stapel wird ein Oberbegriff ausgesucht. Diese Vorgänge geschehen zurzeit jedoch ebenfalls manuell. Einen anderen Ansatz schlägt das Verfahren ein, welches versucht die Einsatzbereitschaft der Benutzer mit einzubinden und damit dem Trend des Web 2.0 zu folgen. Dabei erhält jeder Benutzer die Möglichkeit seine Tags in eine eigene Ordnung zu bringen. Dadurch, dass viele Benutzer die gleichen Tags verwenden und sortieren, entstehen viele verschiedene Ordnungssysteme für ähnliche Tags. Diese können nun verglichen werden. Die häufigsten Gemeinsamkeiten spiegeln die wahrscheinlichste Ordnung wieder. Dieses Prinzip baut also ähnlich wie die Folksonomies auf Selbstorganisation auf, bedarf aber ebenso des Einsatzes der Benutzer.

Neben diesen grundlegenden Optimierungen kann auch der Empfehlungsalgorithmus noch weiterentwickelt werden. Außer acht gelassen wurden eventuell existierende oder in Entwicklung befindliche mathematische Verfahren oder theoretische Herangehensweisen aus der Informatik. Diese könnten noch eine Verbesserung der Ergebnisse bewirken. Ebenso könnte der Algorithmus um eine zeitliche Komponente erweitert werden. Das heißt, dass Tags, welche über einen gewissen Zeitraum nicht benutzt wurden, nicht mehr berücksichtigt werden. Dadurch können alte Objekte, die mit Tags wie „Sommersemester 2004“ gekennzeichnet wurden, aussortiert werden. Denn es ist davon auszugehen, dass dieser Tag in Zukunft nicht mehr benutzt wird und das Objekt nach einigen Jahren für die Benutzer nicht mehr interessant sein wird.

¹⁹⁰ <http://www.google.de/>, letzter Zugriff am 16.07.2009.

¹⁹¹ Vgl. Shmueli, G./ Patel, N./ Bruce, P. (2006), S. 205ff.

5 ZUSAMMENFASSUNG UND AUSBLICK

5.1 Zusammenfassende Betrachtung der Ergebnisse

Das Ziel dieser Entwicklung war es, auf bereits vorhandene Daten eine Empfehlung auszugeben. Dabei wurde das Hauptaugenmerk auf die theoretische- und die technische Umsetzbarkeit, mittels neuester semantischer Technologien, gelegt. Dies wurde dadurch erreicht, dass im ersten Schritt die existierenden Systeme evaluiert wurden. Dabei stellte sich heraus, dass insbesondere das Annotieren von Inhalten mittels Tags ein weit verbreitetes und von den Benutzern akzeptiertes Verfahren ist. Es wurden allerdings auch die Nachteile dieser Methode ersichtlich. Nach der Praxisanalyse folgte eine Ausarbeitung der theoretischen Methoden und Techniken, um Daten im Internet semantisch zu annotieren und zu verarbeiten. Die semantischen Technologien in Form von RDF und OWL sind, auf Grund ihrer Empfehlung und Standardisierung vom W3C, eine solide Basis. Die Herausforderung bestand nun darin, die in der Praxis verwendete Technik des Taggens mit einem semantischen Grundgerüst zu verbinden. Dazu wurden verschiedene Vokabulare untersucht aus denen SCOT als Auswahl hervorging, da es verschiedene andere Vokabulare mit einbezieht und seinen Schwerpunkt auf Tags und Tag Clouds legt. Damit war es möglich Inhalte sowohl in Form von Tags auszuzeichnen, als auch ein solides, zukunftsorientiertes, durchdachtes und komplexes Abfrage- und Speicherungssystem zu nutzen, welches für den Empfehlungsalgorithmus einen wichtigen Baustein darstellt. Um die Nachteile, welche das Tag-Prinzip mit sich bringt, auszugleichen, sollte ein Vokabularmanager genutzt werden. In diesem sollte es möglich sein zusätzlich hierarchische Strukturen anzulegen und Tags in verschiedenen Schreibweisen miteinander in Verbindung zu setzen. Eine Analyse bestehender Ontologieeditoren und anderer Werkzeuge zeigte, dass ein solches Programm für die aufgestellten Anforderungen und Vorgaben noch nicht existierte. Daraufhin wurde eine Eigenentwicklung eines Vokabularmanagers umgesetzt. Im letzten Schritt wurde, aufbauend auf den herausgearbeiteten theoretischen Modellen, ein Algorithmus erstellt und in die Lernplattform integriert, welcher dem Benutzer Empfehlungen zu anderen für ihn interessanten Inhalten aufzeigt.

5.2 Ausblick

Der Empfehlungsalgorithmus kann, neben den bisherigen im System genutzten Einsatzgebieten, auf einfache Weise auf weiteren Seiten genutzt werden. Eine Möglichkeit ist die schon entwickelte, aber noch nicht eingesetzte Tag Cloud in der Benutzeransicht. Es werden vom Anwender häufig benutzte Tags angezeigt, um durch diese auf andere Inhalte zu gelangen. Neben den Empfehlungen können mit dem implementierten System verschiedene auf Tags basierte Verbindungsempfehlungen entwickelt werden.

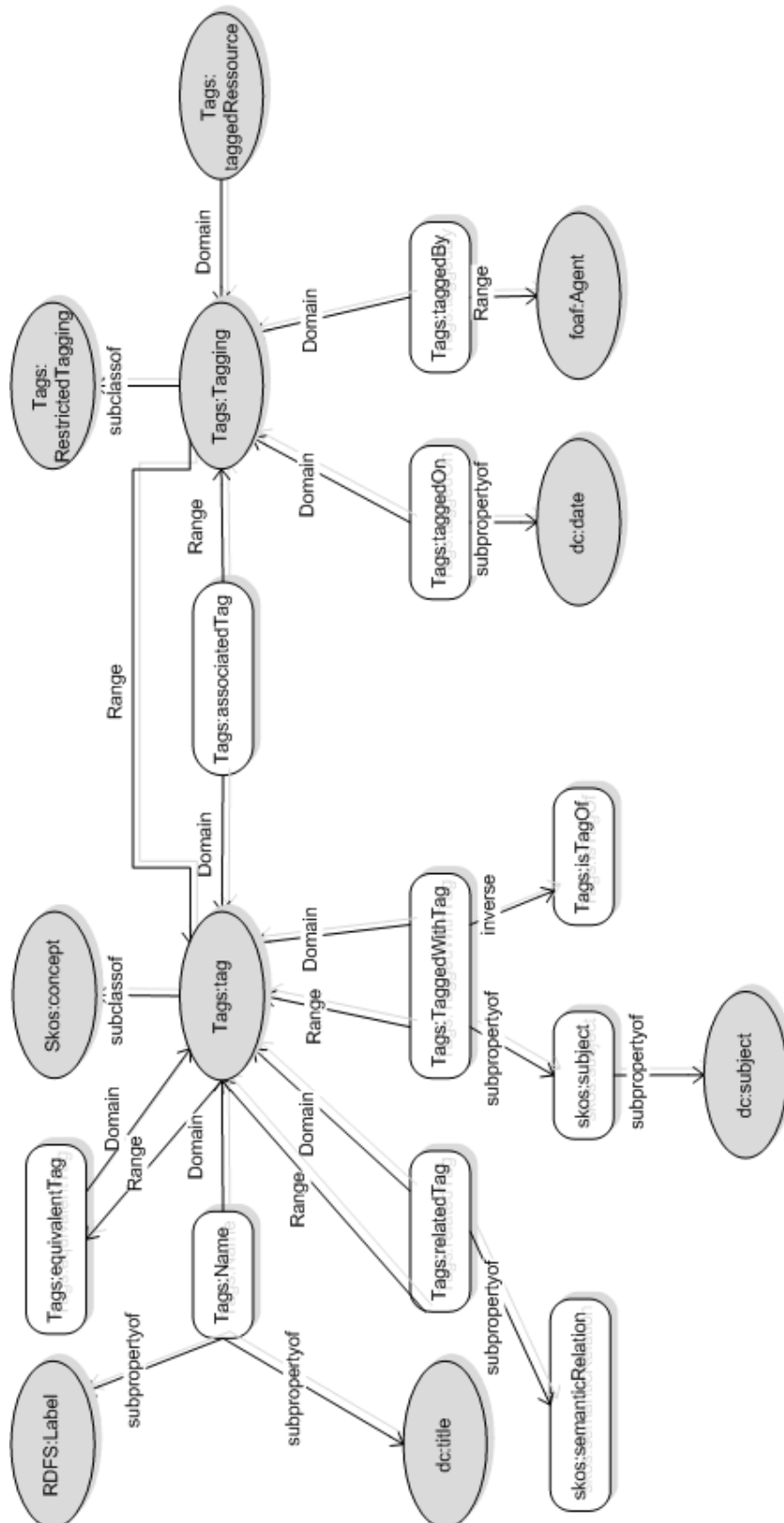
Abseits der Tags bieten insbesondere die semantischen Technologien Erweiterungsmöglichkeiten. Die Auszeichnung verschiedener Inhalte mittels RDFa würde es anderen automatischen Systemen erlauben die Inhalte der Seite auszulesen und kontextuell richtig zu verarbeiten. Als Beispiel ist das

¹⁹² Vgl. Rosenfeld / Morville (2002), S. 235ff.

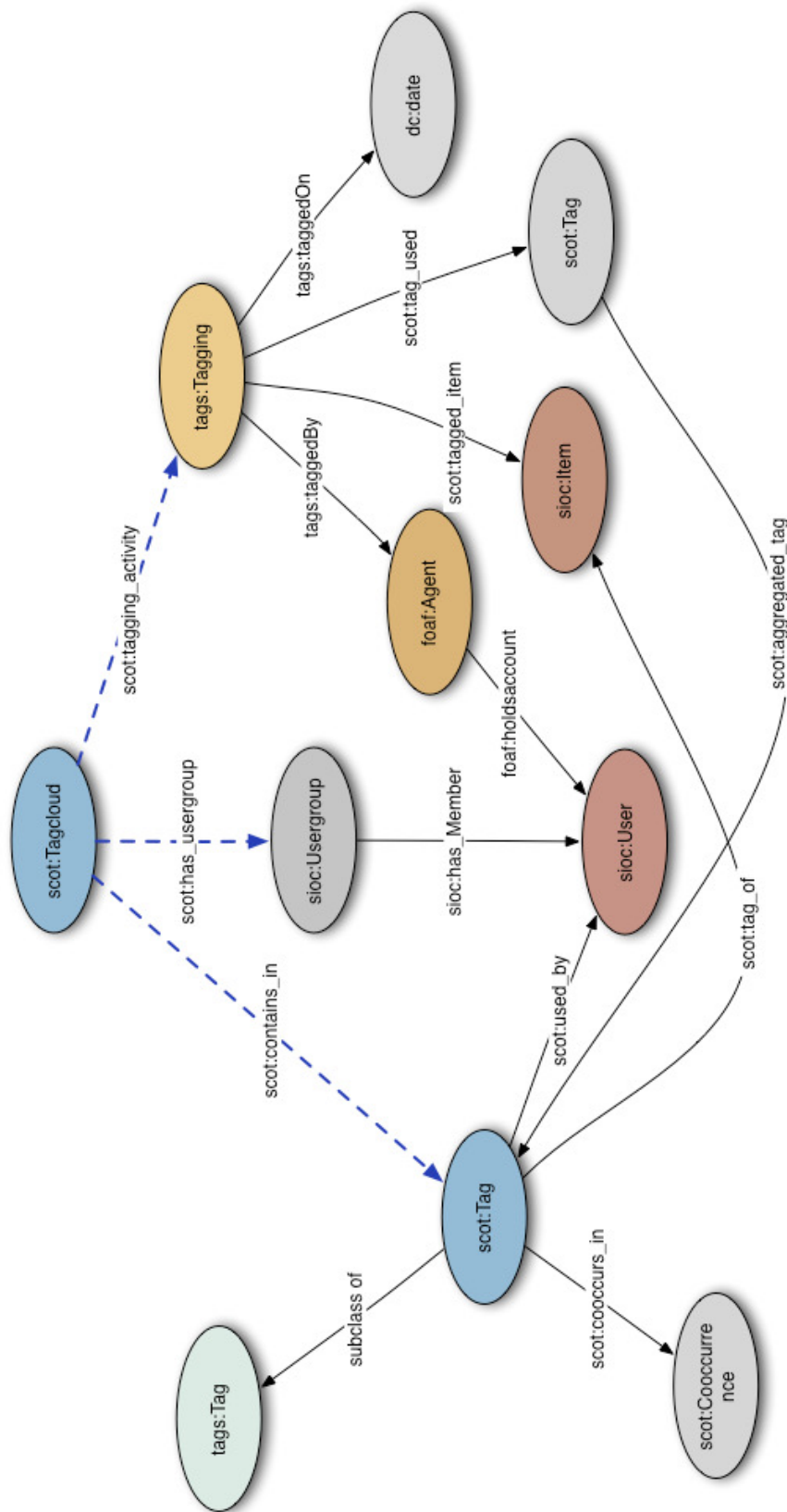
von Yahoo! gestartete Projekt Search Monkey zu nennen, welches solche Inhalte in die Ergebnisse der Suchmaschine mit einbezieht. Sobald die Inhalte einer Webseite maschinell ausgelesen werden und in einen Kontext gebracht werden können, ist es auch möglich Empfehlungssysteme zu erstellen, welche nicht nur auf eine Webseite beschränkt sind, sondern das gesamte Internet mit einbeziehen und dem Benutzer entsprechend seiner Eigenschaften Vorschläge generieren.

ANHANG A

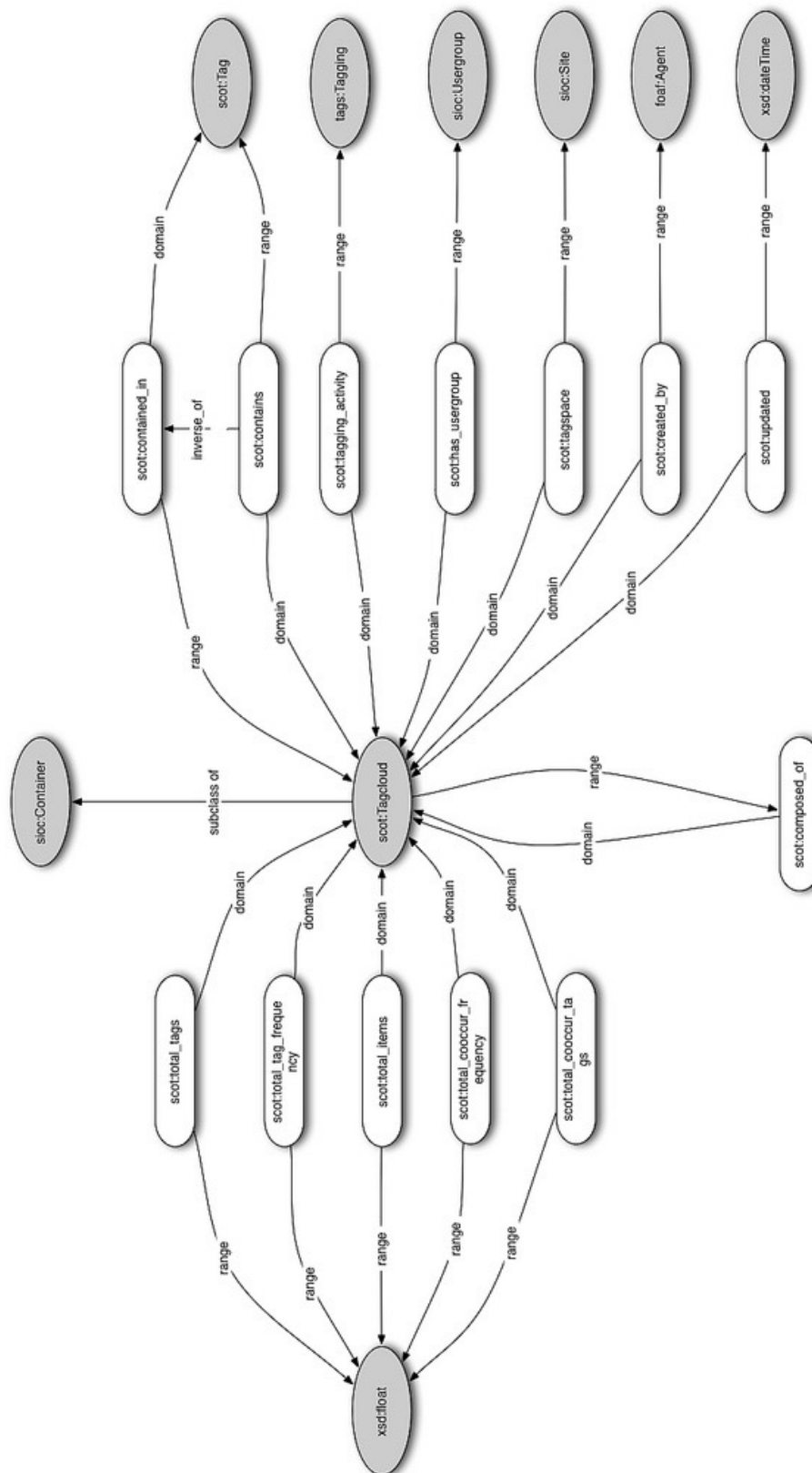
A.1: Richard Newman's Tag Ontology



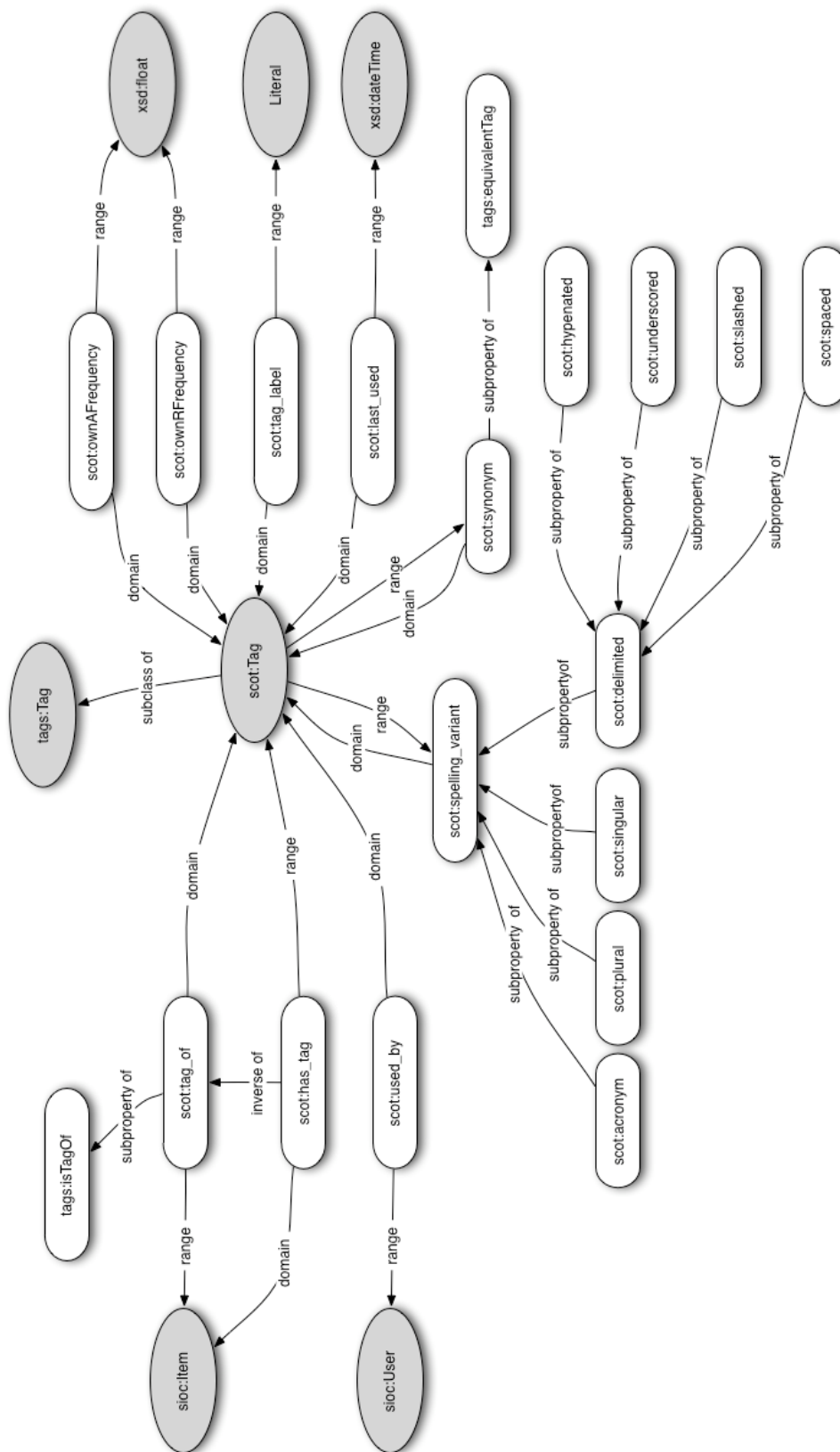
A.2 SCOT – Übersicht



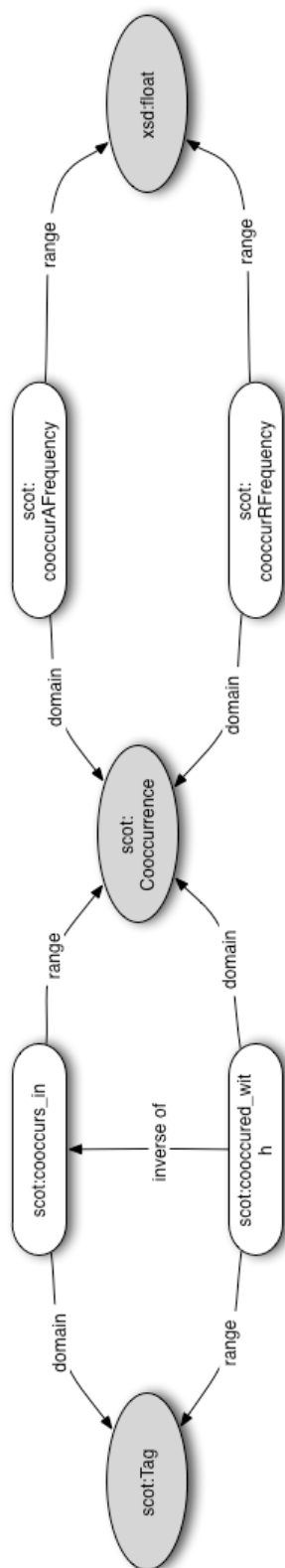
A.3: SCOT – Tag Cloud



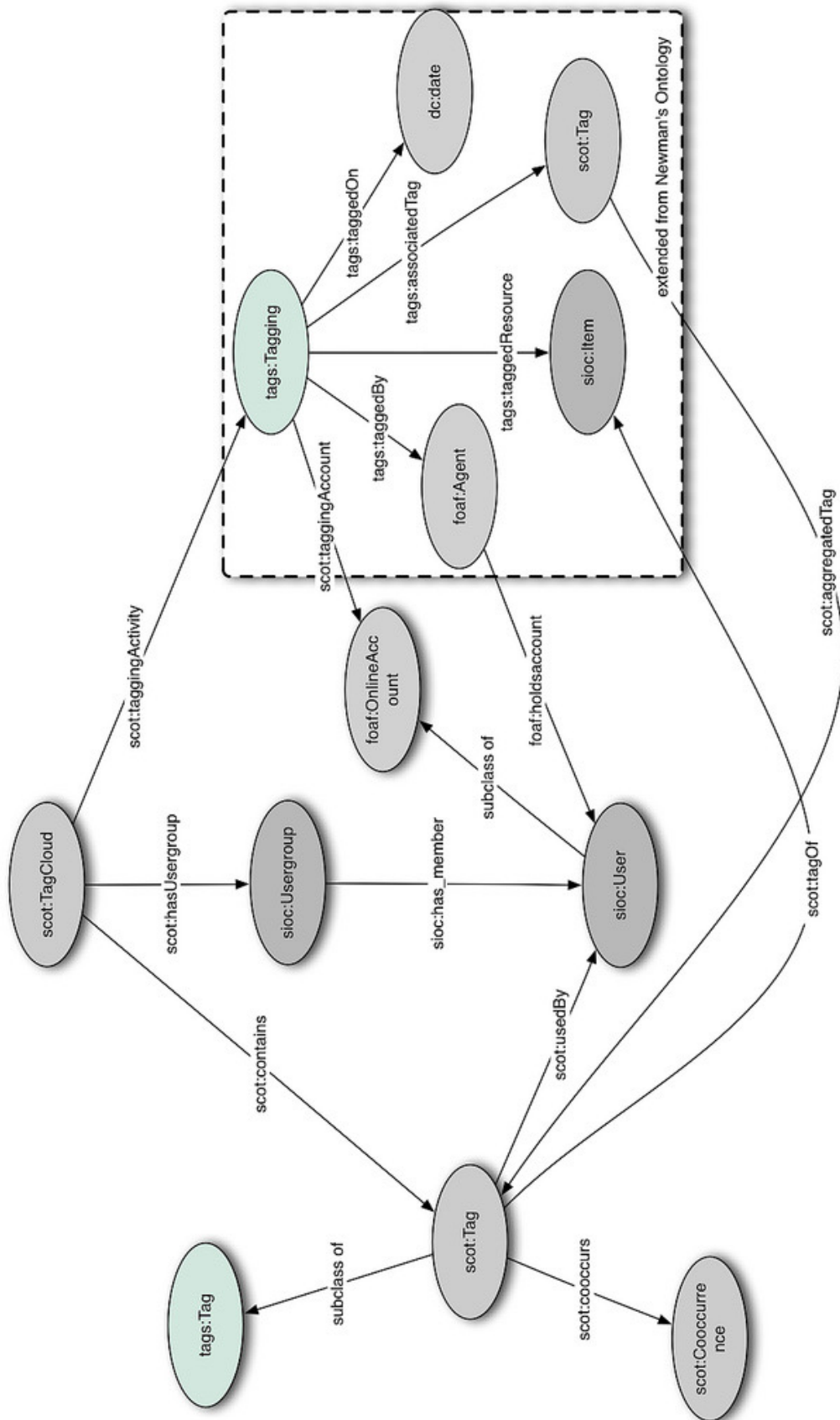
A.4: SCOT – Tag



A.5: SCOT – Cooccurrence



A.6: SCOT – Tagging Activity



ANHANG B

B.1 STUDENTSERVICE GmbH



Die **STUDENTSERVICE** GmbH hat es sich zum Ziel gesetzt Studierenden, Auszubildenden und Schülern zu helfen, in dem es die Möglichkeiten zur Wissensbeschaffung und zum Wissensaustausch verbessert. Dabei liegt der Fokus in Form von Dienstleistungen und Unterstützer didaktischer Konzepte auf dem Bildungssektor. Daraus leitet sich auch das Leitmotiv ab: „bildung . einfach . effizient“. Mit ihren zwei Mitarbeitern und weiteren studentischen Hilfskräften betreibt das Unternehmen verschiedene Projekte:

In Kooperation mit dem DS&OR-Lehrstuhl von Frau Prof. Dr. Leena Suhl an der Universität Paderborn entstand das Interdisziplinäre Projekt, welches Studierende und Unternehmen zusammenbringt. Die Studierenden bekommen die Möglichkeit im Rahmen einer Lehrveranstaltung eigenständig Projekte in Kooperation mit großen Firmen aus dem Wirtschaftsstandort OWL durchzuführen und somit Wissen praxisorientiert anzuwenden.

Darüber hinaus wurden verschiedenste Weblösungen zur Verknüpfung von Wissenschaft und Wirtschaft entwickelt. Ein Dienstleistungsprodukt ist die Bereitstellung einer webbasierten Jobbörse, in der Unternehmen Stellenanzeigen veröffentlichen können. Der Mehrwert dieser Lösung liegt neben der einfachen Handhabung in der Möglichkeit dieses Portal als Applet in jegliche Webseite einzubinden.

Den Wissensaustausch der Studierenden soll das Portal **STUDYLIFE** fördern. Die **STUDENTSERVICE** GmbH fungiert als technischer Bereitsteller dieses Portals für das Studentenwerk Paderborn, welches mit dem Portal ihrem sozialen Auftrag an den Studierenden nachkommt. Unterstützt durch die Hochschulgruppe **STUDYLIFE** Paderborn e.V. werden Downloads, Erfahrungsberichte und ein Forum angeboten. Abseits des reinen Bildungsspektrums finden sich dort außerdem verschiedene Börsen für Jobs, Wohnungen, Kleinanzeigen und Mitfahrgelegenheiten. Für das Studentenleben existieren verschiedene Live-Veranstaltungen wie Tischfußballmeisterschaften oder die Quiznight.

Das dritte Portal ist die Lernplattform learnuzz, auf der Studierende, Auszubildende und Schüler Prüfungsfragen in Form von Multiple-Choice einstellen und gemeinsam mit ihren Kommilitonen beantworten können. Dadurch wird den Prüflingen eine Plattform für das kollaborative Lernen bereitgestellt, wodurch die Klausurvorbereitung nach dem Motto: „study . together . mobile“ vereinfacht wird.

GLOSSAR

- Agent:** Auch Software-Agent, bezeichnet ein Programm mit eigenständigem Verhalten. Agenten werden benutzt um Routineaufgaben, wie die Suche nach bestimmten Inhalten im → Web, zu erledigen.
- ARPAnet:** Vorgänger des Internets, welcher im Jahre 1966 erfunden wurde. Die Organisation ARPA wurde 1957 als Forschunginitiative des Verteidigungsministeriums der USA gegründet.
- API:** Abkürzung für „Application Programming Interface“, eine Schnittstelle zur Programmierung. Siehe auch → Web Service
- Atom:** Dies ist ein XML-Format, das die Nachfolge von RSS antreten soll.
- Blog:** Kurzform von → Weblog.
- CERN:** Kernforschungszentrum mit Sitz in Genf
- CMS:** Abkürzung für Content Management System. Eine Software, die es dem Benutzer erlaubt, Inhalte im Web zu publizieren, ohne dass es dafür → HTML oder eine andere Sprache erlernt werden müsste.
- Community:** Neudeutsch für Gemeinschaft; manche sehen mehr Bedeutung in dem englischen oder deutschen Begriff, die durch die Übersetzung nicht abgedeckt wird.
- CSS:** Abkürzung für Cascading Style Sheet. Eine Sprache, mit der bestimmt wird, wie spezifizierte Inhalte dargestellt werden sollen.
- Digitale Signatur:** Verfahren zur Gewährleistung der Echtheit eines elektronischen Dokuments und der Authentizität des Absenders.
- E-Commerce:** Electronic Commerce; im engeren Sinne: elektronischer Handel; im weiteren Sinne: elektronisch unterstützte Geschäftstätigkeit E-Business.
- Feed:** Dienst, der Inhalte (z.B. News) zum Abonnement bereitstellt. Feeds basieren beispielsweise auf → RSS oder → Atom und kommen vor allem auf Plattformen mit regelmäßig neuen Inhalten zum Einsatz, so dass die Abonnenten die Plattform nicht immer besuchen müssen, sondern bei Änderungen und Neuigkeiten automatisch darüber informiert werden.
- Folksonomy:** Kofferwort aus engl. *folk* (Volk) und *taxonomy* (Taxonomie). Eine durch → Social Tagging erstellte Sammlung von Tags bzw. ein System zur gemeinschaftlichen Ka-

tegorisierung von Objekten, bei dem die Objekte mit frei wählbaren Tags versehen werden.

Framework: Software, die Funktionen und Programmstrukturen bereitstellt, die von allen Anwendungen einer bestimmten Klasse genutzt werden können. Sie erleichtern häufig die Programmierung durch die Bereitstellung häufig genutzter Funktionalitäten.

GRDDL: Kurz für Gleaning Resource Descriptions from Dialects of Languages. Daten können mittels GRDDL aus einer Webseite ausgelesen und als → RDF Tripel dargestellt werden.

HTML: HyperText Markup Language. Auszeichnungssprache zur Darstellung von Inhalten wie Texten, Bildern und Links in → Hypertext-Dokumenten.

HTTP: HyperText Transfer Protocol. Ein Protokoll zur Übertragung von Daten über ein Netzwerk, das hauptsächlich eingesetzt wird, um Webressourcen (insbes. → Hypertext-Dokumente) in einen Webbrowser zu laden.

Hyperlink: Hervorgehobene Text- oder Bildstelle, die mit der Maus angeklickt werden kann und zu einer neuen Datenquelle bzw. Webseite führt. Wird oft nur als Link bezeichnet.

Hypertext: Textform, die im Gegensatz zu linearen Texten eine netzartige Struktur aufweist, welche eine logische Verbindung zwischen verschiedenen Wissenseinheiten herstellt.

Informationsüberflutung: Zustand vom Vorhandensein vielfältiger Informationen zu einem bestimmten Thema (*Information overload*). Durch diese vorherrschende Komplexität können die Nutzer (User) nicht mehr über die Wichtigkeit, Relevanz und Korrektheit der Information entscheiden.

Kollektive Intelligenz: Die Entscheidungen Einzelner können falsch sein, doch wenn die Entscheidungen vieler Menschen zusammengekommen werden, so kann eine bessere Entscheidung entstehen, insbesondere wenn diese Menschen miteinander kommunizieren können.

Netzeffekt: Ein extremer positiver Effekt, der beschreibt, dass der Nutzen an einem Standard oder Netzwerk wächst, wenn dessen Nutzerzahl größer wird. Auch: Netzwerkeffekt.

- Microformats:** Eine Auszeichnungstechnik um Inhalte im Internet maschinenlesbar und – interpretierbar zu gestalten.
- MySQL:** Eine einfache und kostenlose auf → SQL aufbauende Datenbank.
- NFSnet:** Nachfolger des → ARPAnet, aber nur für zivile Zwecke entworfen
- Ontologie:** Konzeptionelle Formalisierung Wissensbereichen in der Informatik. Konkrete Umsetzungen hiervon werden als → Vokabulare bezeichnet.
- Open Source:** Quelloffenheit. Eine Open Source-Software ermöglicht jedem den Einblick in den Quelltext des Programms sowie die Erlaubnis diesen Quellcode beliebig weiterzugeben oder zu verändern.
- OWL:** Abkürzung für „Web Ontology Language“. Basiert auf → RDF, ermöglicht darüber hinaus logische Zusammenhänge zu definieren
- PHP:** Rekursives Backronym für (PHP) Hypertext Preprocessor. Eine serverseitige → Scriptsprache, die hauptsächlich zur Erstellung von dynamischen Webseiten bzw. Webanwendungen verwendet wird.
- Plug-in:** Software, die die Funktionalität eines Webbrowsers um Funktionen erweitert, um bspw. Videos, bestimmte Dateiformate oder → Java-Applikationen anzeigen bzw. ausführen lassen.
- RDF:** Eine vom → W3C standardisierte, formale Sprache zur Bereitstellung von Metadaten im → Semantic Web. Grundidee ist, Eigenschaften von Webressourcen in einer maschinell verarbeitbaren Form zu beschreiben. RDF wird erweitert durch → RDF-Schema, eine Sprache, mit der Beziehungen zwischen Ressourcen definiert werden können.
- RDFa:** Ähnlich wie microformats ein auf → RDF basierendes Auszeichnungsformat um Inhalte in → HTML zu kennzeichnen.
- RDF-Schema:** RDF-Schema ist eine Erweiterung des RDF → Vokabulares
- RSS:** Real Simple Syndication. Eine Technologie, die es ermöglicht, Web-Inhalte (z.B. Text- oder Bildinformationen) zu abonnieren und als → Feed auf Computer, Mobiltelefone oder andere Endgeräte zu laden.
- Semantic Web:** Eine Erweiterung des WWW um maschinenlesbare Daten, die die Semantik der Inhalte formal festlegen. Das Konzept beruht auf einen Vorschlag von WWW-Begründer Tim Berner-Lee.

- semantic web:** In dieser Schreibweise steht der Begriff für die Annotation von Inhalten durch → microformats. Die Kleinschreibung drückt aus, dass diese Form der Umsetzung nicht so mächtig ist.
- Social Network:** Dies sind Netzwerke in denen sich Benutzer im Internet zusammenschließen und kommunizieren können.
- Social Tagging:** Form der Indexierung, bei der tags (Schlagwörter) durch eine größere Gruppe und mit Hilfe verschiedener Arten von → Social Software zugeordnet werden.
- SPARQL:** Abkürzung für SPARQL Protocol and RDF Query Language, ist eine Abfragesprache für Daten, welche im → RDF gespeichert wurden.
- SWRL:** Abkürzung für Semantic Web Rule Language. Durch SWRL wird → OWL um wichtige Funktionen ergänzt, die es erlauben komplexere Schlussfolgerungen abzuleiten.
- SQL:** Structured English Query Language. Eine deklarative Datenbanksprache für relationale Datenbanken.
- Tag:** Schlagwort, mit dem Inhalte assoziiert werden. Im Gegensatz zur starren Kategorisierung können Inhalte mit mehreren Tags versehen werden, so dass sie nicht länger nur einer einzelnen Kategorie zugeordnet sind. Dadurch wird die Suche nach Inhalten vereinfacht. Ein Tag kann auch aus mehreren Wörtern bestehen.
- Tagging:** Dies ist der Vorgang, ein Objekt mit einem → Tag zu versehen.
- Tag Cloud:** Eine Visualisierung verwendeter → Tags, meistens durch die Gewichtung von Tags, zum Beispiel indem häufig verwendete Tags größer dargestellt werden als selten verwendete Tags.
- Traffic:** Der Verkehr auf einer Webseite, gemessen z.B. an der Anzahl der → Visits.
- URL:** Uniform Resource Locator. Ein Uniform Resource Identifier (URI), der Webressourcen über einen primären Zugriffsmechanismus (häufig → http oder FTP) und die Adresse der Ressource identifiziert.
- Usability:** Gebrauchstauglichkeit. Bezeichnet die Eignung eines Produktes bei der Nutzung durch bestimmte Benutzer in einem bestimmten Kontext, die vorgegebenen Ziele effektiv, effizient und zufriedenstellend zu erreichen.
- User-generated Content:** Von Nutzern erstellte Inhalte. Inhalte werden nicht länger lediglich (redaktionell) von Website-Betreibern bereitgestellt, sondern jeder Nutzer be-

kommt die Möglichkeit, eigene Inhalte zu hinterlegen. Dies können z.B. Bewertungen, → Tags oder → Blog-Einträge sein.

Vokabulare: Eine Zusammenstellung von Bezeichnern mit klar abgegrenzter Bedeutung und eindeutigen URLs.

W3C: World Wide Web Consortium, eine Organisation, welche die internationalen Standards für das Internet definiert.

Web: Abkürzung für → World Wide Web.

Web 2.0: Bezeichnet ein Web mit → User-generated Content.

Web 3.0: Dies ist ein anderer Begriff für → Semantic Web.

Web Service: Softwareanwendung, die eindeutig identifizierbar ist und über definierte Schnittstellen beschrieben und gefunden werden kann. Ein Web Service unterstützt die direkte Interaktion mit anderen Anwendungen unter Verwendung → XML-basierter Nachrichten und internetbasierte Protokolle.

Webanwendung: Ein Programm, das auf einem Webserver ausgeführt wird, wobei eine Interaktion mit dem Benutzer ausschließlich über einen Webbrowser erfolgt. Auch: Web-system, Webapplikation.

Weblog: Digitales Tagebuch, das im Internet veröffentlicht wird und in der Regel themenbezogene Einträge beinhaltet, welche von den Lesern mit Kommentaren versehen werden kann.

Wiki: Eine im Web verfügbare, auf → Hypertexten basierende Datensammlung zu verschiedensten Themen, die von Webnutzern nicht nur gelesen, sondern auch online verändert werden kann.

World Wide Web: Auch als WWW, Web oder W3 bezeichnet. Ein auf → Hypertext basierendes System mit grafischer Benutzeroberfläche zum Auffinden und Bereitstellen von Ressourcen im Internet.

XHTML: Extensible HyperText Markup Language. Eine Neuformulierung von → HTML, die sich an die → XML-Syntax hält.

XML: eXtensible Markup Language. Vom W3C definierter Standard zur Modellierung von strukturierten Daten in Form einer Baumstruktur.

µf: Abkürzung für → microformats

Quelle: Die Glossareinträge wurden entnommen aus:

Alby (2007), S. 18ff, 203-220,
Fritz (2004), S. 333-342,
Kampmann (2001), S. 5-8,
Hitzler/ Krötzsch/ Rudolph/ Sure (2008),
Maurice (2009),
Blumauer/ Pellegrini (2009),
Grütter (2006),
<http://www.w3.org/>, letzter Zugriff am 16.07.2009,
<http://www.mysql.de/>, letzter Zugriff am 16.07.2009

LITERATURVERZEICHNIS

- Alby, T. (2007):** Web 2.0 Konzepte, Anwendungen, Technologien, Carl Hanser Verlag, München, 2007.
- Allsopp, J. (2007):** Microformats: Empowering Your Markup for Web 2.0. Apress Verlag Berkeley, 2007.
- Blumauer, A./ Pellegrini, T. (2006):** Semantic Web: Wege zur vernetzten Wissensgesellschaft, Springer-Verlag, Berlin / Heidelberg, 2006.
- Blumauer, A./ Pellegrini, T. (2009):** Social Semantic Web: Web 2.0 – Was nun? Springer-Verlag, Berlin Heidelberg, 2009.
- Breitman, K./ Casanova, M./ Truszkowski, W. (2007):** Semantic Web: Concepts, Technologies and Applications. Springer-Verlag, Berlin, 2007.
- Ebersbach, A./ Glaser, M./ et al. (2008):** Wiki: Web Collaboration. Springer-Verlag, Berlin, 2008, Auflage: 2nd completely rev. and enl. ed.
- Fritz, W. (2004):** Internet-Marketing und Electronic Commerce. Grundlagen- Rahmenbedingungen- Instrumente. Gabler Verlag, Wiesbaden, 3.Auflage, 2004.
- Gaiser, B.[Hrsg.] (2008):** Good Tags – Bad Tags: Social Tagging in der Wissensorganisation. In: Medien in der Wissenschaft; Band 47. Waxmann Verlag, Münster, 2008.
- Grigoris, A./ Harmelen, F. (2008):** A semantic Web primer. MIT Cambridge, 2008.
- Grütter, R. (2006):** Software-Agenten im Web. In: Informatik Spektrum, 2006, Heft 1, Springer-Verlag, Berlin / Heidelberg
- Hitzler, P./ Krötzsch, M./ Rudolph, S./ Sure, Y. (2008):** Semantik Web. Springer-Verlag, Berlin Heidelberg, 2008.
- Maurice, F. (2009):** Microformats : Semantik für Webseiten. Entwickler.press München, 2009.
- Münz, S./ Nefzger, W. (1999):** HTML-4.0-Referenz. Franzis-Verlag, Poing, 1999.
- Pellegrini, T. (2008):** Grundlagen des Semantic Web: Fortschritt mit (r)evolutionärem Potential. In: t3n Nr. 14, 12/2008 – 02/2009, yeebase Verlag, Hannover, 2009.
- Rosenfeld, L./ Morville, P. (2002):** Information architecture for the World Wide Web. O'Reilly,

Beijing, 2006.

Shmueli, G./ Patel, N./ Bruce, P. (2006): Data Mining for Business Intelligence, First Edition.

Wiley-Interscience, Hoboken, NJ, 2006.

Wing Ng, K. (1993): Algorithms and computation. In: 4th International Symposium, ISAAC '93,

Hong Kong, December 1993 : proceedings. Springer-Verlag, Hong Kong, 1993.

INTERNETQUELLENVERZEICHNIS

Adewumi, D. (2008): Facebook offers friend recommendations, another way to keep itself relevant. Verfügbar unter: <http://venturebeat.com/2008/03/26/facebook-offers-friend-recommendations-another-way-to-make-itself-relevant-to/>, letzter Zugriff am 16.07.2009.

Adida, B./ Ayers, D./ et al. (2007): GRDDL Use Cases: Scenarios of extracting RDF data from XML Documents. Verfügbar unter: <http://www.w3.org/TR/grddl-scenarios/>, letzter Zugriff am 16.07.2009.

Adida, B./ Birbeck, M. (2008): RDFa Primer. Verfügbar unter: <http://www.w3.org/TR/xhtml-rdfa-primer/>, letzter Zugriff am 16.07.2009.

Berners-Lee, T. (1998): Semantic Web Road map. Verfügbar unter: <http://www.w3.org/DesignIssues/Semantic.html>, letzter Zugriff am 16.07.2009.

Berners-Lee, T. (2007): Tim Berners-Lee on the Semantic Web. Verfügbar unter: <http://www.technologyreview.com/Infotech/18451/>, letzter Zugriff am 16.07.2009.

Bettel S. (2007): Erste Knoten für das Semantic Web. Verfügbar unter: <http://futurezone.orf.at/stories/230131/>, letzter Zugriff am 16.07.2009.

Cranor, L./ Langheinrich, M./ et al. (2002): The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. Verfügbar unter: <http://www.w3.org/TR/P3P>, letzter Zugriff am 16.07.2009.

DCMI (2005): Using Dublin Core. Verfügbar unter: <http://www.dublincore.org/documents/usage-guide/#basicprinciples>, letzter Zugriff am 16.07.2009.

DCMI (2008): Dublin Core Metadata Element Set, Version 1.1. Verfügbar unter: <http://dublincore.org/documents/dces/>, letzter Zugriff am 16.07.2009.

Hardy, E. (1993): The History of the Net. Verfügbar unter: http://w2.eff.org/Net_culture/net.history.txt, letzter Zugriff am 16.07.2009.

Frodl, C./ Fischer, T. / et al. (2007): Deutsche Übersetzung des Dublin-Core-Metadaten-Elemente-Sets, Version 1.1. Verfügbar unter: http://www.kim-forum.org/material/pdf/uebersetzung_dcmes_20070822.pdf, letzter Zugriff am 16.07.2009.

GMX (2009): Finanzierung, Basisdienste, Werbung, warum? Verfügbar unter:

<http://faq.gmx.de/start/6.html>, letzter Zugriff am 16.07.2009.

Gotter, L. (2006): Text Mining - Wissensgewinnung aus Texten. Verfügbar unter:

<http://wissensexploration.de/textmining-hintergrundwissen.php>, , letzter Zugriff am 16.07.2009.

Hildenbrand, T./ Rothlauf, F./ Heinzl, A. (2006): Ansätze zur kollaborativen Softwareerstellung.

In: – Working Papers in Information Systems University of Mannheim, S. 3. Verfügbar unter:
<http://wi.bwl.uni-mainz.de/Dateien/Arbeitspapier-2006-06-Kollaboration-HiRH06.pdf>, letzter Zugriff am 16.07.2009.

Institut für Software-Ergonomie und Usability (2006): Ungleiche Beteiligung: Wie man mehr Nutzer zu Beiträgen ermutigt. Verfügbar unter: <http://www.usability.ch/Alertbox/20061009.htm>, letzter Zugriff am 16.07.2009.

Isaac, A./ Summers, E. (2009): SKOS Simple Knowledge Organization System Primer. Verfügbar unter: <http://www.w3.org/TR/skos-primer>, letzter Zugriff am 16.07.2009.

Jacobs, B. (2005): Richtlinien zur Erstellung von einfachen Multiple-Choice-Aufgaben nach Gronlund. Verfügbar unter: <http://www.phil.uni-sb.de/FR/Medienzentrum/verweise/psych/aufgaben/mcguideline.html>, letzter Zugriff am 16.07.2009.

Kampmann, M. (2001): Die Geschichte des Internet: Vom ARPAnet bis zum Internetboom. Verfügbar unter: <http://www.techfak.uni-bielefeld.de/ags/wbski/lehre/digiSA/Gedankengeschichte/Ausarbeitungen/2906.pdf>, letzter Zugriff am 16.07.2009.

Kim, A./ Hoffman, L./ Martin, C. (2002): Building Privacy into the Semantic Web: An Ontology Needed Now. Verfügbar unter: <http://semanticweb2002.aifb.uni-karlsruhe.de/proceedings/Position/kim2.pdf>, letzter Zugriff am 16.07.2009.

Kim, H./ Breslin, J./ et al. (2008): SCOT Ontology Specification. Verfügbar unter: <http://scot-project.org/scot/index.html>, letzter Zugriff am 16.07.2009.

Klein, J. (2006): Standart: Plädoyer wider die Besserwisser. Verfügbar unter: <http://www.kfaktor.com/standart/>, letzter Zugriff am 16.07.2009.

Kuhn, J. (2007): Zocker, Zirkus, Dreistigkeit. Verfügbar unter:

<http://www.spiegel.de/wirtschaft/0,1518,470879,00.html>, letzter Zugriff am 16.07.2009.

Linden, G./Smith, B./York, J. (2003): Amazon.com Recommendations: Item-to-Item Collaborative

Filtering. Verfügbar unter:

http://dsonline.computer.org/portal/site/dsonline/menuitem.9ed3d9924aeb0dcd82ccc6716bbe36ec/index.jsp?&pName=dso_level1&path=dsonline/2003_Archives/0301/d&file=wp1lind.xml&xsl=article.xsl&, letzter Zugriff am 16.07.2009.

Lischka, K. (2007): StudiVZ setzt auf Schnüffel-Werbung. Verfügbar unter:

<http://www.spiegel.de/netzwelt/web/0,1518,523286,00.html>, letzter Zugriff am 16.07.2009.

Manola, F./ Miller, E. (2004): RDF Primer. Verfügbar unter: [http://www.w3.org/TR/rdf-](http://www.w3.org/TR/rdf-primer/#dublincore)

[primer/#dublincore](http://www.w3.org/TR/rdf-primer/#dublincore), letzter Zugriff am 16.07.2009.

Microformats (2009): Wiki der microformats Homepage. Verfügbar unter:

http://microformats.org/wiki/Main_Page, <http://microformats.org/wiki/hcard> und <http://microformats.org/about/>, letzter Zugriff am 16.07.2009.

Mikhalenko, P.(2005): Introducing SKOS. Verfügbar unter:

<http://www.xml.com/pub/a/2005/06/22/skos.html>, letzter Zugriff am 16.07.2009.

Newman, R. (2005): Tag ontology design. Verfügbar unter:

<http://www.holygoat.co.uk/projects/tags/>, letzter Zugriff am 16.07.2009.

O'Hara (2004): Trust Strategies for the Semantic Web in Proceedings 3rd International Semantic

Web Conference (ISWC), Workshop on Trust, Security, and Reputation on the Semantic Web, Hiroshima, Japan, 2004. Verfügbar unter: <http://eprints.actors.org/372/01/ISWC04-OHara-final.pdf>, letzter Zugriff am 16.07.2009.

Olbrich, M. (1998): Relationenorientiertes Modellieren mit Objekten in der Bauinformatik. Ver-

fügbar unter: <http://www.bauinf.uni-hannover.de/publikationen/olb1998prom.pdf>, letzter Zugriff am 16.07.2009.

Prodromou, E. (2008): RDFa vs microformats. Verfügbar unter:

http://evan.prodromou.name/RDFa_vs_microformats, letzter Zugriff am 16.07.2009.

Schütt, P (2006): Social Computing im Web 2.0. In: wissensmanagement 3/06, S. 30-33. Verfügbar

unter: http://www.wissensmanagement.net/insight/peter_schuett/2006_03_008.pdf, letzter Zugriff am 16.07.2009.

Schwarz, M. (2008): Süddeutsche gegen SEO. Verfügbar unter: [http://meingottundmeinewelt.de/](http://meingottundmeinewelt.de/2008/06/14/sueddeutsche-gegen-seo/)

[2008/06/14/sueddeutsche-gegen-seo/](http://meingottundmeinewelt.de/2008/06/14/sueddeutsche-gegen-seo/), letzter Zugriff am 16.07.2009.

Semantic Web School (2006): Semantische Technologien und Datenschutz. Verfügbar unter:

http://www.semantic-web.at/newsletter/nl_14feb_06/#0, letzter Zugriff am 16.07.2009.

Stern (2004): Wikipedia schlägt Brockhaus. Verfügbar unter: [http://www.stern.de/computer-](http://www.stern.de/computer-technik/internet/:stern-Test-Wikipedia-Brockhaus/604423.html)

[technik/internet/:stern-Test-Wikipedia-Brockhaus/604423.html](http://www.stern.de/computer-technik/internet/:stern-Test-Wikipedia-Brockhaus/604423.html), letzter Zugriff am 16.07.2009.

Tenbrock, C. (2000): Die Geldvernichter, In: Die Zeit. Verfügbar unter:

http://www.zeit.de/2000/20/200020.online-pleiten_.xml, letzter Zugriff am 16.07.2009.

Vander Wal, T. (2007): Folksonomy. Verfügbar unter:

<http://www.vanderwal.net/folksonomy.html>, letzter Zugriff am 16.07.2009.

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Diplomarbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt und die benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Diese Arbeit liegt in gleicher oder ähnlicher Form keiner Prüfungsbehörde vor.

Paderborn, den

Michael Düppjohann