

Semantic Web und Multimedia.

Untersuchung und Konzeption eines auf OWL basierenden
Retrievalsystems

Bachelorarbeit

an der
HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN
HAMBURG

Fakultät Design, Medien, Information
Studiendepartment Information

vorgelegt von

Peter Schindler

Hamburg, August 2008

Erstreferent: Prof. Dr. Franziskus Geeb
Zweitreferent: Prof. Dr. Martin Gennis

Danksagung

Besonderer Dank gebührt meinem Erstreferenten Prof. Dr. Franziskus Geeb, der mir das Thema vorgeschlagen hat und mich während der Bearbeitungszeit konstruktiv betreut hat. Des Weiteren möchte ich meinem Zweitreferenten Prof. Dr. Martin Gennis herzlich danken. Außerdem gebührt meiner Verlobten, Silvia Strötgen, großer Dank, die mir eine besondere Motivation in der Zeit der Bearbeitung war.

Meiner Mutter und meinem zu früh verstorbenen Vater danke ich, dass sie mir das Studium durch ihren unermüdlichen Einsatz ermöglicht haben.

Abstract

Diese Bachelorarbeit soll eine Übersicht über die Möglichkeiten des semantischen Web in Bezug auf das Retrieval multimedialer Daten, besonders Foto- und Videodateien, geben. Neben einer Zusammenfassung und Erläuterung der wichtigsten Standards in den Bereichen Semantic Web, Inhaltsextraktion und Information Retrieval, wird im praktischen Teil ein Konzept zum Retrieval von multimedialen Daten auf Basis der Web Ontology Language (OWL) erstellt und präsentiert.

Schlagworte:

Semantic Web, Ontologie, Multimedia, Information Retrieval, OWL

Inhaltsverzeichnis

Abbildungsverzeichnis.....	VII
Tabellenverzeichnis	IX
1. Einleitung	1
1.1 Relevanz des Themas	1
1.2 Forschungsthese	3
1.3 Vorgangsweise	3
2. Grundlagen des Semantic Web	5
2.1 Begriffsdefinition Semantik und Syntax.....	5
2.2 Definition Semantic Web.....	8
2.3 Fortschritt und aktueller Stand der Technik.....	10
2.4 Das Schichtenmodell	14
2.5 Ontologien.....	18
3. Metadaten und Werkzeuge	20
3.1 XML	21
3.2 Dublin Core.....	25
3.3 RDF/RDFS	26
3.4 OWL	34
3.5 MPEG-7	38
4. Inhaltsextraktion	41
4.1 Inhaltsextraktion alphanumerischer Dokumente.....	41
4.2 Inhaltsextraktion multimedialer Dokumente.....	44
4.2.1 Inhaltsextraktion von Bilddateien	44
4.2.2 Inhaltsextraktion von Videodateien.....	47
5. Retrieval und Suchverhalten.....	49
5.1 Definition, Methoden- und Modellübersicht.....	49
5.1.1 Mengentheoretische Modelle	51
5.1.2 Vektorraum-basiertes und probabilistisches Retrieval	54
5.1.3 Sprachmodelle und Bayesian Retrieval	54
5.2 Retrieval von alphanumerischen Dokumenten	56
5.3 Retrieval von multimedialen Daten.....	56
5.4 Suchverhalten	57
5.4.1 Searching.....	57
5.4.2 Browsing.....	58
5.5 Ergebnisqualität: Precision und Recall.....	59
6. Konzeption eines auf OWL-basierenden Retrievalsystems	60
6.1 Ontologie Retrievalsystem für Privatanwender	62
6.1.1 Anwenderszenario	62

6.1.2	Mögliche technische Realisierung.....	62
6.2	Ontologie Retrievalsystem für professionelle Anwender	64
6.2.1	Anwenderszenario	64
6.2.2	Mögliche technische Realisierung.....	64
6.3	Ontologiemodellierung	65
6.3.1	Allgemeines	65
6.3.2	Umsetzung Anhand einer einfachen Beispielontologie.....	68
6.3.3	Evaluation der Ontologie	72
6.3.4	Umsetzung Anhand einer komplexeren Themenontologie ...	73
6.3.5	Evaluation der Ontologie	78
6.4	Funktionskonzept Inhaltsextraktion	79
6.5	Funktionskonzept Retrieval	82
6.6	Fazit	84
7.	Zusammenfassung und Ausblick	86
	Glossar	X
	Literaturverzeichnis	XII
	Anhang	XXIV
	Einfache Ontologie – Multimediaontologie.owl	XXIV
	Komplexe Ontologie – Politik.owl.....	XXX
	Eidstattliche Versicherung	XXXVI

Abbildungsverzeichnis

Abb. 1 Aufteilung der Semiotik in die drei Teildisziplinen.....	5
Abb. 2 Syntaxfehler in der DOS-Eingabeaufforderung.....	6
Abb. 3 Korrekter syntaktischer Befehl	7
Abb. 4 Ressourcenbeziehungen beginnend beim Beispielbegriff „Tier“	9
Abb. 5 Aktuelles Semantic Web Schichtenmodell.....	14
Abb. 6 Beispiel für ein Statement in RDF	27
Abb. 7 Beispiel für ein Statement mit leerem Knoten in RDF	29
Abb. 8 RDF-Graph mit RDFS-Elementen.....	34
Abb. 9 Inhaltsextraktion eines Fotos	46
Abb. 10 Schematische Darstellung des Information Retrievals.....	50
Abb. 11 Bayesian Retrieval Model.....	55
Abb. 12 Allgemeine Architektur des Retrievalsystems.....	61
Abb. 13 Allgemeines Konzept der Ontologiearchitektur	66
Abb. 14 Superklasse „Metadaten“	68
Abb. 15 Erste Subklassen.....	68
Abb. 16 Weitere abstrakte Subklassen	69
Abb. 17 Konkrete Klassen	69
Abb. 18 Auflistung aller Slots.....	70
Abb. 19 Beispielinstantz	72
Abb. 20 Superklassen Themenontologie	74
Abb. 21 Subklassen der ersten Ebene.....	74
Abb. 22 Weitere Subklassen der Superklasse Politik und der ersten Subklasse Deutschland.....	75

Abb. 23 Instanzbeispiel Teil 1.....	76
Abb. 24 Instanz der Klasse „Bundeskanzler“	76
Abb. 25 Instanzbeispiel Teil 2.....	77
Abb. 26 Instanz der Klasse „Bundeskanzler“ und „ParteivorsitzCDU“	78
Abb. 27 Ausgangsbild von Bundeskanzlerin Angela Merkel.....	79
Abb. 28 Erste Analyse des Ausgangsbildes.....	80
Abb. 29 Suchabfrage für die Fotos von Angela Merkel.....	82
Abb. 30 Ergebnisliste und Inhalt der ersten Instanz.....	83

Tabellenverzeichnis

Tab. 1 Unterschiede zwischen Semantik & Syntaktik.....	7
Tab. 2 RDFS Klassen	32
Tab. 3 RDFS Eigenschaften	33
Tab. 4 Klassen und Slots.....	71

1. Einleitung

Das Internet hat sich im letzten Jahrzehnt zu einem Hort für eine Unmenge von Informationen entwickelt. Alleine die Suchmaschine „Google“ hat einen Index von mehr als 8 Milliarden Seiten, in denen Nutzer nach Herzenslust suchen können (vgl. GOOGLE 2008) und das sind bei weitem nicht alle Webseiten, die existieren. Hinzu kommen nicht indexierte, generierte Webseiten. Dies sind z.B. Nutzerabfragen von Datenbanken (vgl. ISELE 2005, S.14).

Neben konventionellen Webseiten ist das World Wide Web (WWW) darüberhinaus immer mehr zu einer Plattform für das Präsentieren, Tauschen, Archivieren und Suchen von multimedialen Dokumenten geworden. Die Datenbank für lizenzfreie Bilder „pixelio.de“ hat einen Bilderumfang von fast 23.000 (vgl. HEIN 2008) und die Online-Videoplattform „Youtube“ hatte im September 2006 bereits über 100 Mio. Videoclips auf ihren Servern verzeichnet, Tendenz steigend (vgl. WIKIPEDIA – YOUTUBE 2008). Hinzukommt, dass es heutzutage mittlerweile unproblematisch ist, als Privatnutzer multimediale Dokumente nicht nur selbst zu erstellen, sondern diese auch innerhalb von wenigen Minuten im WWW zu veröffentlichen. Seien es nun mit der eigenen Digitalkamera aufgenommene Fotos, mit dem Handy erstellte Videos oder mit einem einfachen Mikrofon erzeugte Tondokumente. Das WWW bietet für sämtliche multimedialen Erzeugnisse passende Möglichkeiten, diese für viele mögliche Zielgruppen zu veröffentlichen. Das Konzept des „Web 2.0“ macht es möglich.

1.1 Relevanz des Themas

Die bereits erwähnten umfangreichen Möglichkeiten des Veröffentlichens von multimedialen Inhalt schafft durchaus auch neue, große Herausforderungen. Ein Beispiel: Während das Veröffentlichen von

multimedialen Inhalten in den letzten Jahren deutlich leichter geworden ist, ist das Suchen und Finden von bestimmten Dokumenten sehr viel schwieriger geworden. Einfache syntaktische Suchmethoden stoßen bei den Ummengen von Informationen vermehrt an ihre Grenzen. Vor allem wenn der Inhalt des Dokuments unzureichend beschrieben wurde oder einfach nur einen unpassenden Titel enthält, ist die Wahrscheinlichkeit hoch, dass das Dokument, welches man sucht, schlichtweg nicht findet. Ein konkreteres Beispiel wäre die Suche nach Videos von Tennisspielern, die während der Matches der Australian Open ihre Schläge am Netz vollführen (vgl. BLANKEN, BLOK ET AL. 2007, S. 2). Eine syntaktische Suche auf der offiziellen Seite „<http://www.ausopen.org>“ wäre damit bereits überfordert und würde keine brauchbaren Ergebnisse liefern. Diese Bachelorarbeit soll daher Möglichkeiten aufzeigen, wie man diese Problematiken mit Hilfe von semantischen Technologien in den Griff bekommen könnte. Semantische Technologien sind ein Teil des sog. Semantic Web. Der Miterfinder des World Wide Web, Tim Berners-Lee, beschreibt das Semantic Web folgendermaßen: „The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation. (BERNERS-LEE, HENDLER ET AL., S. 3f)“ Die Bachelorarbeit soll u.a. einen Ausblick auf die aktuellen Möglichkeiten des Retrievals von multimedialen Inhalten geben und ggf. aufzeigen, wie genau und wo welche Verbesserungen nötig wären.

Die Grenzen dieser Thematik gehen über privates Nutzerinteresse hinaus. Es gibt bereits erste Konzepte, z.B. aus dem Bereich der Bioinformatik. Das Projekt „BioDash¹“ der „W3C Healthcare and Life Sciences Interest Group“ versucht mit Hilfe von Ontologien medizinische Informationen semantisch zu erfassen (vgl. QUAN 2007 S. 173). Es soll ein Zusammenhang mit Hilfe einer Topic Map² erstellt werden, welcher eine Art semantische

¹ Das Wort Dash wird in diesem Fall als Abkürzung des Wortes „Dashboard“ (=Armaturenbrett) verwendet.

² Abstraktes Modell zur Visualisierung von Wissenstrukturen.

Verknüpfung aus mehreren medizinischen und pharmazeutischen Teilbereichen darstellt. (vgl. W3C 2005 – HAYSTACK).

1.2 Forschungsthese

Die Forschungsthese dieser Bachelorthesis lautet: „Die Ontologiesprache OWL hat das Potential, das Suchverhalten sowie die Ergebnisqualität von Suchanfragen bei dem Retrieval von Film- und Fotodaten erheblich zu verbessern“.

Es soll eine Möglichkeit gefunden werden, anhand der Nutzung der Web Ontology Language (OWL) die Grenzen, die bisherige syntaktische Suchsysteme setzen, zu überwinden und eine Basis für eine leistungsfähigere Suchmöglichkeit für Multimedia-Content zu konzipieren. Das Ziel soll sein, ein Konzept zu entwerfen, welches auf der Basis von OWL einen verbesserten Retrievalprozess für Film- und/oder Videodaten darstellt.

1.3 Vorgangsweise

Um ein detailliertes Verständnis von semantischen Anwendungen im Bereich von Multimedia zu bekommen, ist es von hoher Priorität grundlegende Kenntnisse des Semantic Web und von Retrievalverfahren bzw. den dazugehörigen Anwendung zu haben. In Kapitel 2 wird ein Überblick über grundlegende Begriffe zum Thema Semantic Web gegeben. Darüberhinaus soll ein möglichst aktueller Stand der Technik präsentiert werden. Kapitel 3 behandelt das Thema Metadaten, welches relevant für die inhaltliche Erschließung von Daten alphanumerischer sowie multimedialer Art ist. Es werden Beispiele für Auszeichnungssprachen aus dem Gebiet der Inhaltserschließung gegeben, u.a. wird auf die für die These wichtige Ontologiesprache OWL im Detail eingegangen. Das vierte Kapitel geht genauer auf die Inhaltsextraktion von sowohl alphanumerischen, als auch

multimedialen Dokumenten ein. Es werden manuelle, semi-automatische und automatische Verfahren vorgestellt. Im letzten Grundlagenkapitel (Kapitel 5) stehen dann Suchverhalten und Retrieval im Vordergrund. Auch hier soll zwischen alphanumerischen und multimedialen Daten getrennt werden, um die Komplexität bei letzterem zu verdeutlichen. Dies gilt sowohl für die Suche, als auch für die Präsentation der Suchergebnisse. In Kapitel 6 sollen dann die Möglichkeiten untersucht werden, ob die verschiedenen Semantic-Multimedia-Retrieval-Grundlagen miteinander so kombiniert werden können, um eine Anwendung zu konzipieren, die das Retrieval und Suchverhalten erleichtern könnte. Hier ist das Wissen der vorherigen Kapitel essentiell, da auf komplexere Zusammenhänge der Thematik eingegangen wird. Anschließend folgt dann ein mögliches Konzept zum multimedialen Retrieval unter Einbeziehung der Ontologiesprache OWL. Im 7. Kapitel erfolgt eine Schlussbetrachtung. Der Grundlagenanteil dieser Bachelorarbeit entstand anhand des Studiums, Vergleichs und Zusammentragens der Erkenntnisse aus der aktuellen Fachliteratur (Grundlagenmonographien, Projektwebseiten). Da es sich im Bereich Semantic Multimedia Retrieval hauptsächlich um erste Prototypen von Anwendungen und Konzeptionen zum Zeitpunkt der Entstehung dieser Bachelorarbeit handelt, waren die Primärquellen in den Kapiteln in erster Linie (Online-)Zeitschriftenaufsätze. Davon ist hauptsächlich das Kapitel 6 beeinflusst. Das Hauptaugenmerk dieser Untersuchung liegt, wie bereits das Thema treffend beschreibt, im Bereich der multimedialen Dokumente. Jedoch ist es nachvollziehbarer, Inhaltsextraktion und Retrievalverfahren zu den sog. alphanumerischen Dokumenten (Textdokumenten, Tabellen etc.) vergleichend darzustellen, um somit Unterschiede, Gemeinsamkeiten und damit schlussendlich die Komplexität solcher Systeme darzulegen.

2. Grundlagen des Semantic Web

Dieses Kapitel stellt einen ersten Einblick in den Aufbau des semantischen Webs dar. Neben den zwei wichtigen Definitionen für die Begriffe „Semantik“ und „Semantic Web“ soll anschließend die bisherige Entwicklung aufgezeigt werden. Meilensteine des Semantic Web werden beschrieben und der aktuelle Stand der Technik. Eine detailliertere Funktionsbeschreibung erfolgt anhand des allgemeingültigen, aktuellen Schichtenmodells, was eine verständliche Erklärung einzelner Prozesse zulässt. Zum Abschluss des Kapitels steht der Begriff „Ontologie“ im Vordergrund, der ein wichtiges Standbein von semantischen Anwendungen darstellt.

2.1 Begriffsdefinition Semantik und Syntax

Der Begriff „Semantik³“ ist ursprünglich ein Teilgebiet der Sprachwissenschaften. „Zusammen mit der Pragmatik und Syntaktik ist die Semantik eine Unterdisziplin der Semiotik, der allgemeinen Lehre der Zeichen (WIKIPEDIA 2008 – SEMANTIK).“ Zur Verdeutlichung beschreibt die Abb. 1 den Zusammenhang der drei Unterdisziplinen innerhalb der Semiotik.

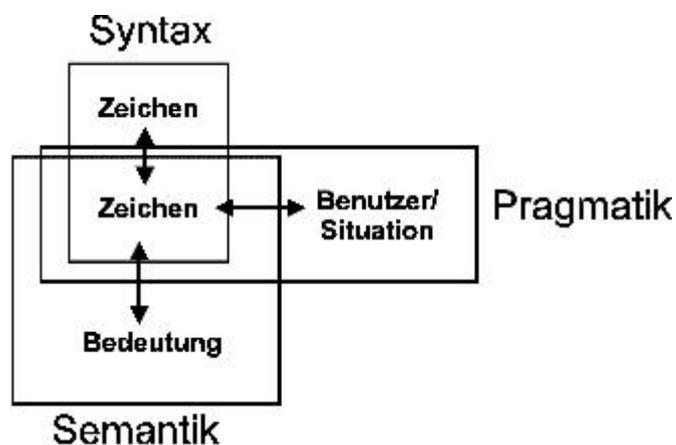
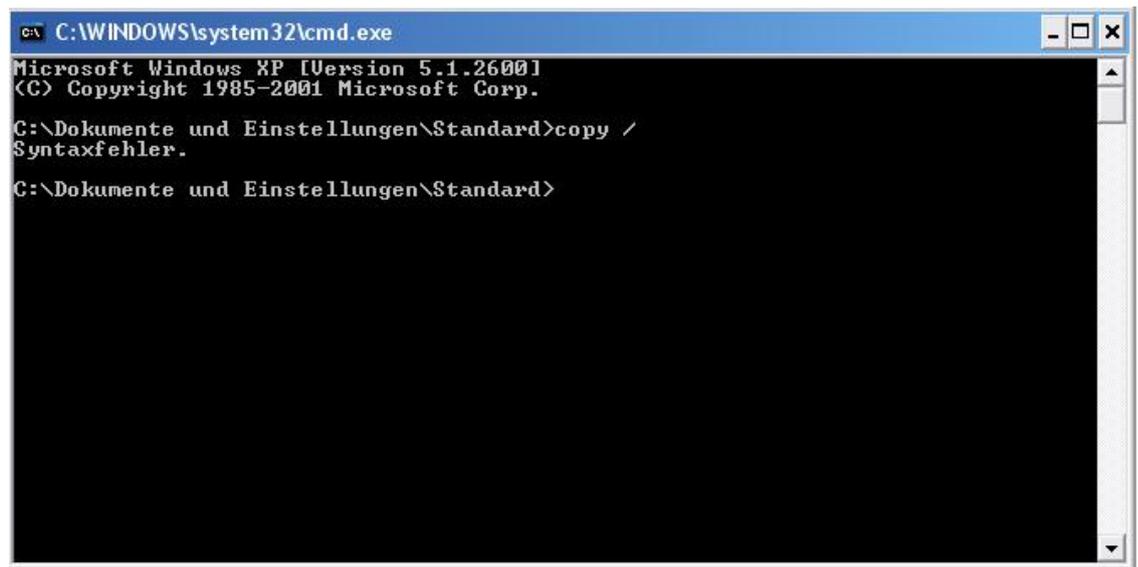


Abb. 1 Aufteilung der Semiotik in die drei Teildisziplinen (WERNER 1997)

³ Wortursprung aus dem Griechischen. σημαίνειν *sēmainein* = bezeichnen, anzeigen.

Im Bereich der theoretischen Informatik wird der Begriff meist als „formale Semantik“ beschrieben. Diese unterscheidet sich vor allem darin, dass mathematische Ansätze verwendet werden (vgl. WIKIPEDIA 2008 – FORMALE SEMANTIK).

Demgegenüber steht der Begriff „Syntax⁴“. Wie in der Sprachwissenschaft beschreibt der Begriff auch in der Informatik die Form und Struktur von Zeichenketten. Abb. 2 und 3, sowie Tab.1 sollen verdeutlichen, wie sich Syntax und Semantik definieren und unterscheiden können.

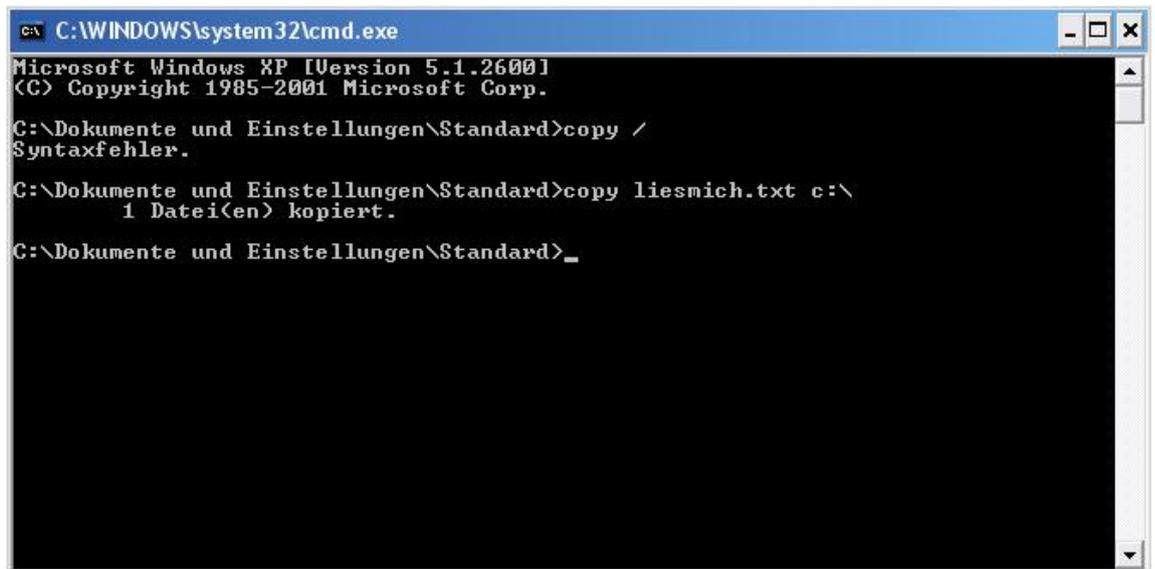


```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\Dokumente und Einstellungen\Standard>copy /
Syntaxfehler.
C:\Dokumente und Einstellungen\Standard>
```

Abb. 2 Syntaxfehler in der DOS-Eingabeaufforderung

In der DOS-Eingabeaufforderung soll der Befehl „Copy“ ausgeführt werden. Dies scheitert jedoch an einem Syntaxfehler. Das Programm versteht den Befehl nur, wenn es einen syntaktisch richtigen Befehl bekommt. In diesem Beispiel ist der Befehl „copy /“ kein syntaktisch richtiger Befehl. Ein syntaktisch richtiger Befehl besteht beim Programm „copy“ aus dem aufgerufenem Program, einer zu kopierenden Datei und einem Zielverzeichnis. Abb. 3 zeigt schließlich, wie eine korrekter syntaktischer Befehl aussehen könnte.

⁴ Wortursprung aus dem Griechischen. σύνταξις = die Zusammenstellung.



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\Dokumente und Einstellungen\Standard>copy /
Syntaxfehler.
C:\Dokumente und Einstellungen\Standard>copy liesnich.txt c:\
1 Datei(en) kopiert.
C:\Dokumente und Einstellungen\Standard>_
```

Abb. 3 Korrekter syntaktischer Befehl

Eine weitere konkrete Möglichkeit dem Unterschied zwischen Semantik und Syntaktik innerhalb der Informatik darzustellen, bieten einfache mathematische Aussagen.

<p>Aussage 1 $3 + / =]$ semantisch und syntaktisch falsch</p> <hr/> <hr/>
<p>Aussage 2 $3 + 4 = 34$ syntaktisch richtig, aber semantisch falsch</p> <hr/> <hr/>
<p>Aussage 3 $3 + 4 = 7$ sowohl semantisch, als auch syntaktisch richtig</p>

Tab. 1 Unterschiede zwischen Semantik & Syntaktik

Aussage 1 ist mathematisch grundlegend falsch. Zu der Zahl 3 soll ein undefiniertes Symbol zu addiert werden. Außerdem ist das Ergebnis ebenfalls ein undefiniertes Symbol.

Aussage 2 enthält zwar eine syntaktisch richtige Zeichenfolge im Sinne der Mathematik, ist allerdings rechnerisch und daher auch semantisch falsch, da das Addieren von 3 und 4 nicht 34 ergeben kann.

Aussage 3 dagegen ist sowohl von der Zeichenfolge, als auch von der mathematischen (semantischen) Sicht her, absolut korrekt.

2.2 Definition Semantic Web

Das Semantic Web ist, wie Kapitel 1.1 bereits erwähnt, keine neue Version des World Wide Web. Es soll eine sinnvolle Ergänzung des aktuellen Webs sein, um bestehende Barrieren, vor allem die von aktuellen Retrievalmöglichkeiten, zu überwinden. Tim Berners-Lee, Mitentwickler des World Wide Web, hob den Nutzen eines semantischen Webs hervor, welches die semantischen Bedeutungen erkennt und für den Nutzer so aufbereitet, dass er einen Vorteil gegenüber der Verwendung aktueller Anwendungen hat. Hüseman definiert drei Hauptziele für das Semantic Web:

- „Informationen im Internet auf der Basis von expliziten, wohl definierten Aussagen über die Bedeutung von Information besser zu strukturieren,
- zu Automatisierungszwecken, zur Applikations-übergreifenden Informationsintegration,
- um den Menschen im Umgang mit Maschinen besser zu unterstützen“ (HÜSEMANN 2005, S. 33).

Die Strukturierung der im Web vorhandenen Informationen sind der erste Schritt zu einem semantischen Web. Ein Agent⁵ der mit der Suche nach einem bestimmten Suchgebiet beauftragt wird, soll eine möglichst genaue und richtige Verbindung zwischen verwandten Begriffen finden, um u.U. weitere für den Nutzer interessante Informationen anbieten zu können bzw. irrelevante Informationen zu filtern. Die Funktionsweise solch eines semantischen Agenten könnte auf dem Prinzip der Wissensrepräsentation basieren (vgl. WIKIPEDIA 2008 – SEMANTISCHES WEB). Innerhalb eines bestimmten Themengebietes sollen Ressourcen⁶ daher in einer eindeutigen Beziehung stehen. Abb. 4 zeigt das Prinzip einer Ressourcenbeziehung.

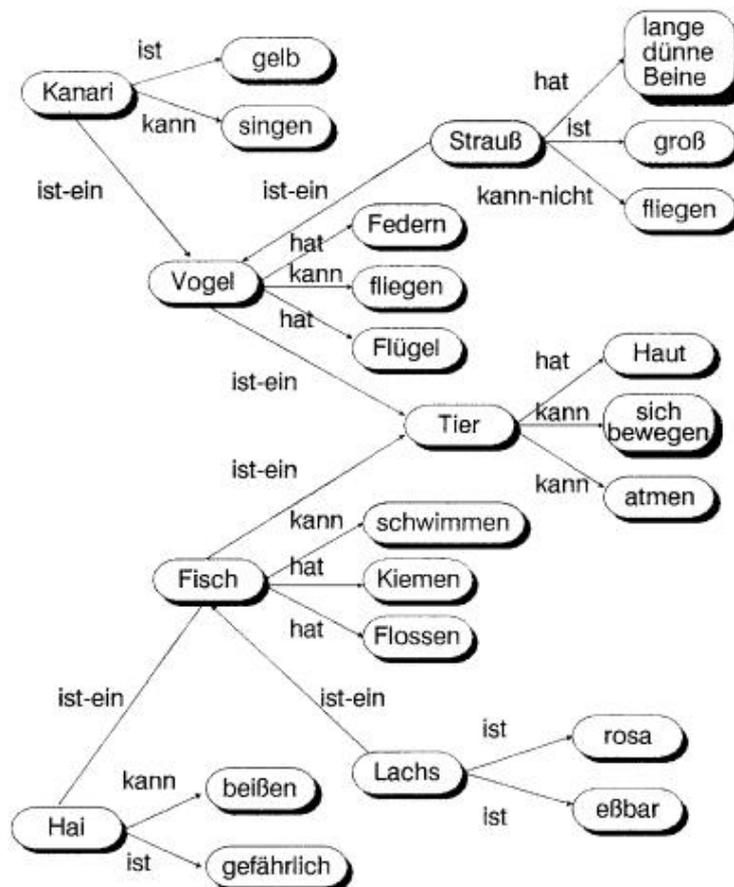


Abb. 4 Ressourcenbeziehungen beginnend beim Beispielbegriff „Tier“ (WAGNER 2006)

⁵ In diesem Fall ein Programm zur Suche von Webseiten, nach vom Nutzer bestimmten Kriterien.

⁶ In diesem Fall könnten es Webseiten, Suchbegriffe, Fotos, Videos etc. sein.

Solche Modelle der Wissensrepräsentation können durchaus simpler, aber auch deutlich komplexer sein. Berners-Lee definiert daher die Herausforderung des Semantic Web in diesem Gebiet, dass sämtliche Daten und Regeln solcher Modelle in eine Auszeichnungssprache für das Web integriert werden sollen (vgl. BERNERS-LEE, HENDLER ET AL. 2001, S. 6). Im Falle einer Webseite muss dessen Erzeugung über den HTML⁷-Quelltext hinausgehen. Eine Kodierung mit HTML dient ausschließlich Web Browsern zur Interpretation der Formatierung der einzelnen Seitenabschnitte. Jedoch kann kein Suchsystem und kein Programm schlussfolgern, welcher Textteil z.B. zum Impressum gehört. Die Strukturierung der Daten auf einer Webseite gehen bei der Gestaltung mit HTML verloren, und als Folge dessen ist der Inhalt für Maschinen ausschließlich als unstrukturierter Text analysierbar (vgl. HÜSEMANN 2005, S. 31).

2.3 Fortschritt und aktueller Stand der Technik

Ohne zu sehr vorzugreifen, wie weit der Stand des Semantic Web im Bereich der multimedialen Anwendungen bereits heute ist, soll ein Blick unabhängig auf aktuelle Projekte und Applikationen geworfen werden. Es soll einen kleinen Überblick geben, inwieweit es bereits heute Fortschritte im Sinne des Nutzers gibt und wie diese konkret aussehen.

Isele definiert in seiner Diplomarbeit den „Sinn und Nutzen des Semantischen Webs“ mit Hinblick sowohl auf die Business- als auch auf die Privatanwender. Ein erwähntes Beispiel, in dem ein Nutzer eine Reise bucht und der entsprechende Agent diese nicht nur bucht, sondern auch „spezifische Informationen“ für den Nutzer heraussucht, und diese dann individuell darstellt, ist heutzutage in dem Umfang noch nicht realisierbar (vgl. ISELE 2005, S. 15f). In so einem Fall wird heutzutage bislang relativ

⁷ Hypertext Markup Language, Standardsprache zur Gestaltung von Webseiten.

viel Entscheidungsbereitschaft des Nutzers in Bezug auf einen Onlinebuchungsvorgang abverlangt.

Obwohl es bis heute noch nicht gelungen ist, Applikationen mit einem so umfassenden semantischen „Können“ auszustatten, gibt es tatsächlich erste Projekte, die in der Lage sind einige semantische Kriterien zu erfüllen:

Swoogle

Die Suchmaschine Swoogle ist eine von der eBiquity research group an der Universität Maryland entwickelte semantische Suchmaschine, die es möglich macht u.a. Metadaten zu Webdokumenten und ganze Ontologien⁸ zu bestimmten Themenbereichen zu finden. Bislang sind bereits über 10.000 Ontologien indexiert (vgl. EBQUIITY RESEARCH GROUP (HG.) 2007). Swoogle versucht anhand einer Unterteilung der gesuchten Dokumente in sog. SWO⁹ und SWDB¹⁰. Es werden Relevanzverhältnisse errechnet und Metadaten für die Dokumente erstellt, um Beziehungen zu erörtern und dann in das Suchergebnis mit einzubauen (vgl. FREIE UNIVERSITÄT BERLIN 2007 – SWOOGLE).

Semantic Desktop¹¹

Beim Begriff Semantic Desktop handelt es nicht um eine konkrete Semantic Web Applikation, sondern viel mehr um ein Konzept zur Integration von semantischen Technologien für den alltäglichen Umgang mit PC Systemen. Es sollen mit dem Einsatz von Definitionssprachen des Semantic Web Dokumente erweitert werden, damit Software inhaltliche Schlüsse aus den Dokumenten ziehen kann und somit u.U. Verknüpfungen erstellt. Im Grunde soll eine effizientere Arbeitsorganisation die Folge des Einsatzes der Semantic Desktop Applikationen sein (vgl. LAURIERE 2007).

⁸ S. Kapitel 2.5.

⁹ SWO = SemanticWebOntology.

¹⁰ SWDB = SemanticWebDatabase Definition: „Dabei ist alles was keine Definitionen oder Erweiterungen von Konzepten besitzt ein SWDB. (FREIE UNIVERSITÄT BERLIN 2007 – SWOOGLE).“

¹¹ Desktop = Arbeitsoberfläche.

Beispiel für eine Applikation aus dem Bereich Semantic Desktop ist u.a. die Wissensmanagement Software DeepaMehta, die es ermöglicht innerhalb von Dokumenten die genannten inhaltlichen Verknüpfungen und Beziehungen zu erstellen (vgl. RICHTER 2008).

Semantic Wiki

Semantische Wikis orientieren sich, wie z.B. Semantic Desktop Applikationen an einem höheren Nutzen durch Verwendung von Metadaten unter Einbeziehung von semantischen Definitionssprachen. Dr. Sebastian Schaffert, wissenschaftlicher Leiter des NewMediaLab in Salzburg definiert die Ziele eines semantischen Wikis folgendermaßen: „Die Vorteile eines Semantischen Wikis liegen einerseits in verbesserter, ‘intelligenter’ Suche und Navigation und andererseits in der Möglichkeit, die Darstellung und Bearbeitung des Inhalts an den Kontext anzupassen“ (SCHAFFERT 2007 - SEMANTISCHE WIKIS HABEN HOHEN NUTZEN ALS WERKZEUG, UM LEHRE UND FORSCHUNG BESSER ZU ORGANISIEREN).

Ziel eines solchen Wikis wäre demnach auch, das gesammelte und verknüpfte Wissen durch automatische Schlussfolgerungen erweitern zu können. IkeWiki ist ein Beispiel für ein semantisches Wiki. Es enthält aktuelle Semantic Web Standards, es verwertet Metadaten zu den jeweiligen Dokumenten und es bietet auch einen unkomplizierten Zugang für Nutzer mit wenig Erfahrung im Umgang mit traditionellen Wikis (vgl. SCHAFFERT 2006 – IKEWIKI, S. 393ff).

FOAF

Das Friend-Of-A-Friend (FOAF) Projekt wurde im Jahre 2000 gegründet, um ein maschinenlesbares soziales Netzwerk zu erschaffen. Ausgangspunkt ist die Veröffentlichung einiger persönlicher Informationen (z.B. Name, Nachname, E-Mail Adresse) in einer RDF¹²-Datei. Abhängig von der Quantität der Veröffentlichungen besteht dann auch in diesem Fall die

¹² Resource Description Framework s. Kapitel 3.3.

Möglichkeit von Verknüpfungen der Dokumente und damit der einzelnen Personen. Ziel ist es, möglichst viele dieser Verknüpfungen zu nutzen, um die Kommunikation zu verbessern, indem man verschiedene Dienste mit anderen Usern (Friends) teilt (vgl. SULLIVAN 2008 – THE FRIEND OF A FRIEND).

Agricultural Ontology Service

Angeregt wurde das Projekt im Jahre 2000 von der Welternährungsorganisation (FAO), welches Ontologien erschaffen sollte, die im Gegensatz zu den bisherigen Thesauri eine einheitlich und umfassende Struktur beinhalten sollte. Dies sollte ein Suchen und Finden innerhalb der Terminologien deutlich vereinfachen (vgl. WIKIPEDIA 2008 - AGRICULTURAL ONTOLOGY SERVICE). Grundlage dieses Projekts ist die Web Ontology Language¹³ (vgl. LIANG, LAUSER ET AL. 2006 – FROM AGROVOC TO THE AGRICULTURAL).

SemanticGov

Urheber dieses Projekts ist ein Konsortium aus mehreren europäischen Behörden, Universitäten und Firmen (insgesamt 11 Organisationen aus 7 europäischen Staaten). Der volle Titel des Projektes lautet: „Providing Integrated Public Services to Citizens at the National and Pan-European level with the use of Emerging Semantic Web Technologies.“ Außer den Vorteilen für die einfachen Bürger soll auch eine Steigerung der Effizienz innerhalb der behördlichen Arbeit ermöglicht werden. Alles in allem verfügt das Projekt über ein Gesamtbudget von fast 4,4 Mio. €, wovon ca. 2,7 Mio. € aus EU-Mitteln finanziert werden (vgl. TARABANIS 2006).

¹³ OWL s. Kapitel 3.4 f.

2.4 Das Schichtenmodell

Das Semantic Web Schichtenmodell¹⁴ (vgl. HERMAN 2008 – W3C SEMANTIC WEB ACTIVITY) wurde vom W3C veröffentlicht, um auf eine anschauliche Art und Weise darzustellen, wie die verschiedenen Technologien aufeinander aufbauen und mit einander in Relation stehen. Eine Gewährleistung, dass eine Anwendung auf einer semantischen Basis die komplette Leistungsfähigkeit erbringt, hängt davon ab, inwieweit diese das Schichtenmodell mit einbezieht und realisiert. Es ist allerdings auch möglich eine Semantic Web Applikation zu erstellen, welche nur auf einen Teil des Modells ausgelegt ist (vgl. ISELE 2005, S. 17f).

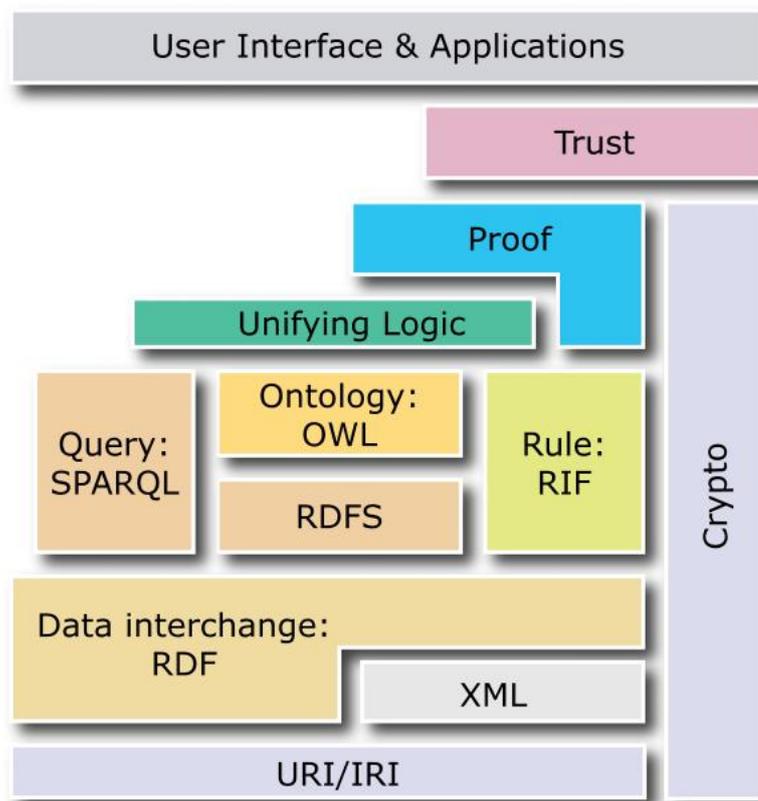


Abb. 5 Aktuelles Semantic Web Schichtenmodell (HERMAN 2008)

¹⁴ Engl. Layer Cake Diagram.

Es wird angemerkt, dass in diesem Unterkapitel ausschließlich die Schichten „URI/IRI“, „Crypto“, „Query: SPARQL“, „Rule RIF“, „Unifying Logic“, „Proof“, „Trust“ und „User Interface & Applications“ näher erläutert werden. Die Schichten, die einen Großteil des Mittelbaus des Modells ausmachen, werden aus Gründen der Gliederung dieser Bachelorarbeit erst zum Ende dieses bzw. im nächsten Kapitel genauer behandelt.

URI/IRI

Basis einer jeden Semantic Web Anwendung ist die Beschreibung von Ressourcen. Der Begriff Ressource wird im Themenbereich des Semantic Web für die beschreibende Elemente, wie z.B. Webseiten, E-Mails, Bilder, Videos etc. bzw. „a thing we want talk about“ (KOMPATSIARIS, HOBSON 2008, S. 29). Um eine Ressource eindeutig zu identifizieren, benötigt es einen „Bezeichner“. URI bzw. IRI sind solche Bezeichner und dienen diesem Zweck. URIs kommen vielen Nutzern bei ihrer täglichen Nutzung vor. URLs sind Formen von URIs, also die Webadresse <http://www.bui.haw-hamburg.de/> ist eine eindeutige Bezeichnung (URI) der Webseite des Departments Information der HAW Hamburg.

Im Gegensatz zur URI bestehen die Internationalized Resource Identifiers (IRI) aus druckbaren ASCII-Elementen, während die Basis von URIs das Unicode-Zeichenformat¹⁵ ist. Das Konstrukt einer URI besteht aus einem Schemateil und einem, durch einem Doppelpunkt getrennten, schemaspezifischen Teil. Im Falle der Webseite des Departments Information wäre das „http“ der Schemateil und das „www.bui.haw-hamburg.de“, der schemaspezifische Teil.

¹⁵ ISO-Standard, in dem alle Zeichen sämtlicher Schriftkulturen enthalten sind.

Crypto

Die Schicht „Crypto“ beinhaltet zum größten Teil sicherheitsrelevante Konzepte. Die Idee eines sog. „Web of Trust¹⁶“ ist auch u.a. Kern der Crypto-Schicht. Mit Hilfe von digitalen Signaturen soll gewährleistet werden, dass eine gewisse Authentizität vorhanden ist. Andererseits sollen auch Nutzer, die Inhalte publizieren, für diese auch bürgen. Daneben spielt die Nutzungsverwaltung eine wichtige Rolle. Inhalte, die nur einem bestimmten Nutzerkreis erreichen sollen, werden durch bestimmte Lizenzen verteilt und auch nur den gewünschten Personen zugänglich gemacht (vgl. ISELE 2005, S. 36f).

Query: SPARQL

Grundlage eines erfolgreichen Retrievals von Informationen ist eine geeignete Abfrage für die Daten, die durchsucht werden sollen. Wichtigstes Werkzeug hierfür ist eine funktionierende Abfragesprache. Eine solche Abfragesprache ist SPARQL¹⁷. Diese Sprache hat vom W3C bislang noch nicht den Status einer „Recommendation“ erhalten, da es sich noch in einer fortgeschrittenen Entwicklungsphase befindet. Jedoch soll diese in absehbarer Zeit abgeschlossen sein, so dass sie in näherer Zukunft zu den offiziellen Empfehlungen aufgenommen wird (vgl. HITZLER 2008, S. 202). Eine wichtige Besonderheit dieser Abfragesprache ist, im Gegensatz zu z.B. SQL¹⁸ oder XQuery¹⁹, dass sie nicht nur auf der Basis, der für das Semantic Web nötigen Definitionssprache RDF aufbaut, sondern auch Resultate in dieser liefern kann (vgl. HERMAN 2008 – W3C SEMANTIC WEB FAQ).

Rule RIF

Rule RIF ist Thema einer aktuellen W3C Working Group, die sich mit der Entwicklung fester Regelstandards, in Form einer Regelsprache, befasst. Diese soll systemunabhängig funktionieren, so dass die Integration von

¹⁶ Engl. für Netz des Vertrauens.

¹⁷ Abk. für SPARQL Protocol and RDF Query Language.

¹⁸ Abk. für Structured Query Language = Abfragesprache für relationale Datenbanken.

¹⁹ Abk. für XML Query Language = Abfragesprache für XML-Datenbanken.

Daten unterschiedlicher Systeme problemlos realisierbar sein wird (vgl. HERMAN 2008 – W3C SEMANTIC WEB FAQ) .

Unifying Logic

Aufbauend auf den drei Mittelschichten „Query: SPARQL“, „Ontology: OWL“ und Rule: RIF“ befindet sich die Schicht der „Unifying Logic²⁰“ (UL). Die UL soll unter Einbeziehung jener Mittelschichten in der Lage sein z.B. Ressourcen mit einander sinnvoll zu kombinieren, anschließend aus diesen Kombinationen unter Umständen neue Schlüsse ziehen. Ergebnisse sollen anschließend so aufbereitet werden, damit sie von der nächst höheren Schicht erörtert werden können (vgl. ISELE 2005, S. 33).

Proof

Als nächst höhere Schicht soll der Proof²¹-Layer die Aufgaben der Erklärung der erhaltenen Ergebnisse z.B. durch einen beauftragen Suchagenten herleiten. Der Nutzer soll in einer für ihn verständlichen Weise nachvollziehen können, wie die für ihn relevanten zusammengestellt wurden und vor allem, was für Informationen zusammengestellt wurden (vgl. SIZOV 2007, S. 94f). Trotz eines großen Umfangs an Möglichkeiten der zur Verfügung stehenden Methoden, z.B. Algorithmen, Erfahrungswerte, Faustregeln etc. ist ein relevantes Ergebnis nicht immer auffindbar bzw. darstellbar. Suchanfragen, die eine zu großen Prüfumfang umfassen, können durchaus ergebnislos abgebrochen werden. Auch eine plausible Beweisfindung, für die jeweilige Ergebnisdarstellung stellt noch große Problem dar (vgl. ISELE 2005, S. 34f) . Generell kann man sagen, dass diese Schicht noch durchaus Entwicklungspotential aufweist und deshalb noch nicht im vollen Umfang funktioniert.

²⁰ Engl. für vereinigende Logik.

²¹ Engl. für Beweis, Nachweis.

Trust

Neben ersten prüfenden Schichten „Logic“ und „Proof“ enthält das Semantic Web Schichtenmodell einen abschließenden Mechanismus zur Feststellung der „Glaubwürdigkeit“ des erfragten Suchergebnisses. Trotz vieler Unterschiede zu den Funktionsweisen des aktuellen Webs, wird es im Semantic Web im Hinblick auf die Möglichkeit weiterhin eigenes Material zu veröffentlichen, weiterhin keine Beschränkungen geben. Das bedeutet, dass es möglich sein wird, Dokumente online zu stellen, deren Inhalt nicht unbedingt „wahr“ sein muss (vgl. KOIVUNEN, MILLER 2001). Der Trust²²-Layer soll daher als abschließende Instanz die Vertrauenswürdigkeit der dargestellten Informationen genauer analysieren, auch wenn es durch die vorherigen Schichten (Logic und Proof) theoretisch nicht mehr erforderlich sein sollte. Der Trust-Layer macht sich die Verwendung von sog. Vertrauensketten zu Nutze. Durch diese kann abgeleitet werden, wie vertrauensvoll eine Quelle ist. Je mehr Vertrauen Personen einer Quelle entgegenbringen, um so wahrscheinlicher ist, dass der Trust-Layer, diese dann auch als vertrauenswürdig anerkennt (vgl. ISELE 2005, S. 36).

User Interface & Applications

Auf dieser Ebene befindet sich die jeweilige Applikation, also quasi das, was der Nutzer sieht und interagiert. Es könnte sich um ein Suchsystem handeln oder einen Agenten, welcher vom Nutzer für eine Suchanfrage konfiguriert wird. Die Schichten unterhalb dieser betreffen hauptsächlich Entwickler.

2.5 Ontologien

Der Begriff Ontologie²³ ist ursprünglich weder ein Begriff aus dem Bereich des Semantic Web noch aus dem Bereich der Informatik. Er stammt aus der Philosophie und beschäftigt sich mit Fragestellung des „Seienden und der

²² Engl. für Vertrauen.

²³ Entstanden aus den griech. Worten „On“ für sein und „Logos“ für Wort, Lehre.

Wirklichkeit“ (MUNZINGER ONLINE 2007 – ONTOLOGIE). In der Informatik (Wissensmanagement) wird der Begriff Ontologie im Zusammenhang mit einer konkreten Wissensrepräsentation verwendet, der ein vereinbartes Vokabular und geeignete Beziehungsregeln zu Grunde liegen. Ihre Aufgaben definiert Hüseemann folgendermaßen: 1. Kommunikation, 2. automatisches Schließen, 3. Repräsentation (vgl. HÜSEMANN 2005, S. 35f). Hauptsächlich werden Ontologien auf inhaltlich abgrenzende Wissensgebiete (Domains) beschränkt. Diese können dann das spezielle Vokabular, aber auch die Beziehungen untereinander, wie sie z.B. aus Glossaren, Taxonomien oder Thesauri stammen, enthalten. Diese bezeichnet man dann auch als „leichtgewichtige“ Ontologien (lightweight Ontologies), im Gegensatz zu den „schergewichtigen“ Ontologien (heavyweight Ontologies), die zusätzlich noch Axiome enthalten, also Aussagen, die immer wahr sind und nicht aus den enthaltenen Begriffen abzuleiten sind, und Einschränkungen, so dass eine größere Klarheit bei einzelnen Aussagen innerhalb der Ontologie zu erkennen ist. Es gibt Versuche Ontologien zu entwickeln, die sich über mehrere Wissensgebiete erstrecken. Diese haben jedoch mit der Problematik zu kämpfen, dass sie sehr aufwendig zu kombinieren sind (vgl. WIKIPEDIA 2008 – ONTOLOGIE (INFORMATIK)). Der qualitative Wert einer Ontologie ist stark von der Akzeptanz einer jeweiligen Nutzergruppe abhängig. Richtlinien sind daher notwendig, um Ontologien zu entwickeln, die hohen qualitativen Standards genügen. Um ein theoretisches Ontologiemodell in die Praxis umsetzen zu können, benötigt man Ontologiesprachen. Je nach Umfang der zu realisierenden Ontologie, spielt die Auswahl der Sprache eine wichtige Rolle. Eine übersichtliche Ontologie benötigt auch eine eher simple Sprache (RDFS, OWL Lite). Dementsprechend werden für umfangreiche Ontologien, auch Sprachen mit einer höheren Vielfalt an Möglichkeiten benötigt (OWL DL, OWL Full).

3. Metadaten und Werkzeuge

Während das zweite Kapitel mehr oder weniger theoretische Grundlagen und einige Beispiele zum Semantic Web vermitteln sollte, geht dieses dritte Kapitel konkret auf Möglichkeiten für die Realisierung von semantischen Konzepten ein. Um einen weiteren Schritt in Richtung einer Retrievalmöglichkeit für multimediale Dokumente zu machen, benötigt es Kenntnisse im Bereich Metadaten. Metadaten sind systematisch, strukturierte Daten, die Informationen zu anderen Daten enthalten. Ihre Struktur ist ein wesentlicher Schlüssel für die Interpretierbarkeit durch Maschinen. Um jedoch auch eine Einheitlichkeit zu gewährleisten, bedarf es Standards (vgl. YU 2007, S. 10f). Neben diesen Standards sind die jeweiligen Werkzeuge vonnöten, die sich in diesem Kapitel als Definitionssprachen darstellen. Sie machen eine standardisierte Annotation von Dokumenten mit den jeweiligen Metadaten möglich. Der Begriff „Annotation“ (semantische Annotation) bedeutet im Bereich des Semantic Web die Anreicherung von Daten mit Metadaten (vgl. HANDSCHUH, STAAB 2003, S. 4).

Der thematische Einstieg beginnt mit der syntaktischen Grundlage, nämlich der Extensible Markup Language (XML). Anschließend folgt ein einfaches Beispiel für einen Standard im Bereich Webseiten-Annotation (Dublin Core). RDF und RDFS ermöglichen dann, erste simple Beziehungen und kleinere Ontologien zu kreieren. Danach steht das Unterkapitel OWL im Vordergrund, in dem präsentiert wird, wie man umfangreichere Ontologien erstellen kann. OWL spielt für die aufgestellte These dieser Bachelorarbeit eine inhaltlich sehr wichtige Rolle. Zum Abschluss des Kapitels erfolgt noch ein Exkurs in einen Metadatenstandard explizit für multimediale Dokumente (MPEG-7). Dieser ist für die Fachwelt von hoher Bedeutung, da bereits Anwendungen mit diesem Standard existieren und ebenfalls eine Retrievalverbesserung anstreben.

3.1 XML

Die Extensible Markup Language (XML) ist, wie die Sprachen RDF(S) und OWL eine Empfehlung des W3C und dient als Auszeichnungssprache für alle Werkzeuge, die in diesem Kapitel behandelt werden, als syntaktische Basis. Aus diesem Grund wird in diesem Kapitel nur auf die wichtigsten Grundlagen eingegangen, da die Sprachen, RDF(S) bzw. OWL relevanter für die Beantwortung der These sind. Gründe für den Empfehlungsstatus sind folgende: Die Dokumente, die mit XML produziert werden, sind generell unabhängig verständlich für verschiedenste Applikationen und Systeme. Die verwendete Syntax für die entstehenden Metadaten können standardisiert werden. Es ist eine Standardstruktur sowohl für Daten als auch für Dokumente vorhanden und es ist keine grundlegend neue Technologie (vgl. DACONTA, OBRST ET AL. 2003, S. 28). Ebenso wie die Grundlagensprache für die Erstellung von Webseiten HTML²⁴, ist auch XML eine sog. Auszeichnungssprache. Auch HTML dient der Annotation, nämlich der mit Formatierungsangaben für die jeweiligen Informationen einer Webseite. Während jedoch für HTML ein festdefiniertes und begrenztes Vokabular zur Verfügung steht, ist XML für die Vergabe von Namen für die Tags²⁵ (oder auch Elemente) kaum an irgendwelche Zwänge gebunden (vgl. HITZLER 2008, S. 17). Zusätzlich können die Elemente mit Attributen versehen werden. Daneben können die festgelegten Eigenschaften vererbt werden.

²⁴ HTML = Hypertext Markup Language.

²⁵ Engl. Tag für Kennzeichnung. In diesem Fall eine Kennzeichnung für eine Teil eines Markup Dokuments.

Ein Beispiel für den Unterschied zwischen HTML und XML:

HTML-Code:

Dieses Wort soll `fett` geschrieben werden.

Ergebnis:

Dieses Wort soll **fett** geschrieben werden.

XML-Code:

Das Buch hat den Autor `<autor>Max Mustermann</autor>`

Ergebnis:

Die Applikation, die diese Aussage verarbeitet, identifiziert „Max Mustermann“ u.U. tatsächlich als Autoren des Buches.

XML-Code:

`<autor>Max Musterman</autor>` schreibt an einem Buch mit der Thematik `<buch type=“Sachbuch“>Multimediale Anwendungen</buch>`

Ergebnis:

Die Anwendung erkennt hier den Autor und das Buch „Multimediale Anwendungen“ und ordnet es als Sachbuch ein.

Um eine einwandfreie Interpretation des XML-Codes durch dementsprechende Applikationen möglich zu machen, muss das XML-Dokument zwangsläufig wohlgeformt sein. Das bedeutet, dass es z.B. „der definierten Grammatik im Bezug auf Groß- und Kleinschreibung, Verschachtelung usw. exakt folgt“ (ISELE 2005, S. 25). Ist dies nicht der Fall, erhält der Nutzer womöglich eine Fehlermeldung und die Auswertung des Dokuments ist nicht weiter möglich. Es sollte des Weiteren beachtet werden, dass ein Elementname generell nur mit einem Buchstaben oder einem Unterstrich („_“) beginnen kann. Des Weiteren beginnt ein XML-

Dokument mit genau einem Wurzelement, worin sich sämtliche anderen Elemente unterordnen müssen.

Um für ein bestimmtes XML-Dokument ein definiertes Strukturmuster zu verleihen, gibt es die Möglichkeit diese in einer „Document Type Definition“²⁶ (DTD) oder einem XML Schema zu definieren. Sie gewährleisten, dass das XML-Dokument nicht nur wohlgeformt, sondern auch als valide erkannt wird (vgl. DACONTA, OBRST ET AL. 2003, S. 36). In einer DTD z.B. kann festgelegt werden, wie genau einzelne Elemente verschachtelt werden können, welche und wie viele Attribute sie besitzen dürfen und welche Zeichen für die Elemente zulässig sind. DTDs können innerhalb des XML-Dokuments enthalten sein oder aus einer externen .dtd Datei mit eingebunden werden.

Beispiel für eine vereinfachte DTD:

```
<?xml version=1.0“?>
<!DOCTYPE Buch [
  <!ELEMENT Buchtitel (Buchtitelbezeichnung+, Untertitel?)>
  <!ELEMENT Autor (Autor*, Herausgeber*)>
  <!ELEMENT ISBN (ISBN+)>
  ...
]>
<Buch>
...
</Buch>
</xml>
```

Das Beispiel zeigt eine vereinfachte DTD für das Wurzel-Element²⁷ „Buch“ innerhalb eines XML-Dokuments. Es werden weitere Elemente deklariert, wie z.B. die Buchtitelbezeichnung, die durch das „+“ mindestens einmal

²⁶ Engl. Dokumenten Typdefinition.

²⁷ Element, dessen Start-Tag am Beginn des XML-Dokuments stehen muss.

vorkommen muss. Hier könnte z.B. noch festgelegt werden, wie die Buchtitelbezeichnung in einer anderen Sprache lauten würde. Außerdem ist im Element das Element Untertitel mit einem „?“ modifiziert, was bedeutet, dass der Untertitel keinmal oder genau einmal vorkommen soll. Ein „*“, welches das Element Autor modifiziert, impliziert, dass es keinmal oder beliebig oft vorkommen kann. Größere Möglichkeiten der Strukturierung von XML-Dokumenten bieten der XML-Schema-Standard. Es können genaue Informationen über die Anzahl der verschachtelten Elemente angegeben werden, Datentyp-Definitionen, wie z.B. integer²⁸, string²⁹ etc. sind bereits enthalten und die syntaktische Grundlage von XML-Schema ist, anders als bei DTDs, XML. XML-Schema-Dokumente und XML-Dokumente können daher vom selben Programm verarbeitet werden. Dies ist ein weiterer Grund, weshalb die DTDs immer mehr vom XML-Schema abgelöst werden (vgl. HITZLER 2008, S. 24f).

Mehrere XML-Dokumente sind theoretisch ohne weiteres kombinierbar, was allerdings Probleme mit sich bringt. Es kann zu Namenskonflikten kommen, wenn mehrere Dokumente gleiche Elementnamen für verschiedene Dinge vergeben. Um diese Namenskonflikte zu umgehen, gibt es die Möglichkeit sog. Namespaces³⁰ zu definieren. Mit Hilfe einer Zuweisung einer URI kann ein Element eindeutig bezeichnet werden. Die Bildung eines Namespaces besteht aus einem Elementnamen, der Angabe „xmlns“, einem Präfix, welches als Abkürzung für Tags verwendet wird und der verwendeten URI. Namespaces werden generell im Start-Tag des jeweiligen Elements deklariert und sind für dieses dann auch gültig (vgl. BRAY, HOLLANDER et al..2006) .

²⁸ Nichtnegative Ganzzahl.

²⁹ Beliebige Zeichenkette.

³⁰ Engl. Namensräume.

Beispiel für einen Namespace:

```
<Buch xmlns:sachbuch=http://www.example.com/sachbuch>  
< sachbuch:Titel>  
Multimediale Anwendungen  
</sachbuch:Titel>
```

3.2 Dublin Core

Dieses Unterkapitel ist als kurzer Exkurs in die konkrete Anwendung von Metadaten gemeint. Streng genommen ist Dublin Core (DC) kein Bestandteil des Semantic Web, kann aber z.B. mit XML interpretiert und eingebunden werden. Der Dublin Core-Metadatenstandard wurde im Jahre 1995 in Dublin, Ohio von mehreren Bibliothekaren, Archivaren, Vertretern von Geistes- und Naturwissenschaften und Vertretern des immer relevanter werdenden öffentlichen Internets gegründet. Dies geschah innerhalb eines Workshops und unter der Schirmherrschaft des Online Computer Library Centers (OCLC) und des National Center for Supercomputing Applications (NCSA) (vgl. YU 2007, S. 11). Ziel war es einen flexiblen Metadatenstandard zur formalen Erfassung von Webressourcen zu entwickeln (vgl. HOLZHEID 2005, S. 17f). Dublin Core ist vorhanden in einem simple (einfachen) Standard, welcher 15 Elemente enthält und einem qualified (näher bestimmten) Standard, mit drei zusätzlichen Elementen (vgl. HILLMAN 2005). Die enthaltenen Elemente sind allesamt optional und können so häufig wie nötig für die formale Erfassung eines Dokuments verwendet werden. Bei Bedarf können die Elemente um sog. Qualifiers³¹ ergänzt werden.

³¹ Elemente, die sowohl Vorgänger- als auch Nachfolgeelemente semantisch detaillieren können.

Beispiel für DC Metadaten im Head-Bereich eines HTML-Dokuments:

```
<head profile="http://dublincore.org/documents/dcq-html/">
  <title>Dublin Core</title>
  <link rel="schema.DC" href="http://purl.org/dc/elements/1.1/" />
  <meta name="DC.format" scheme="DCTERMS.IMT"
content="text/html" />
  <meta name="DC.publisher" content="Jimmy Whales" />
  <meta name="DC.subject" content="Dublin Core Metadaten-Elemente,
Anwendungen" />
  <meta name="DC.creator" content="Björn G. Kulms" />
  <meta name="DCTERMS.license" scheme="DCTERMS.URI"
content="http://www.gnu.org/copyleft/fdl.html" />
</head>
```

Die Standardelemente bieten u.a. Angaben zum Titel, Verfasser, Herausgeber, aber auch zu inhaltlichen, zeitlichen und lizenzrechtlichen Punkten.

3.3 RDF/RDFS

XML bietet zwar die syntaktische Grundlage für das Semantic Web, allerdings hat es auch seine Grenzen. Zwar ist es eine flexible Sprache, die für Menschen bei einer passenden Tagwahl interessante und nützliche Informationen bietet, jedoch semantische Aussagen für Maschinen nur sehr bedingt erstellen kann (vgl. HITZLER 2008, S.29f). Das Resource Description Framework (RDF) bietet erste Möglichkeiten diese sog. semantische Lücke zumindest zu verkleinern, indem Statements erstellt werden können, die sowohl von Nutzern als auch von Maschinen interpretierbar sind. Für das Semantic Web hat RDF eine vergleichbare Relevanz, wie HTML für das World Wide Web und die dazugehörige Erstellung von Webseiten (vgl. YU 2007, S. 40).

In RDF werden Statements in sog. Graphen anschaulich dargestellt. Das einfachste Statement besteht aus einer Ressource, also einem zu beschreibenden Element, einer Property, einer Ressource, die etwas aussagt und einem dazugehörigen Literal. Allgemein lässt sich so ein Statement in

Subjekt, Prädikat und Objekt untergliedern. Ressource und Property müssen eindeutig benannt werden, was mit Hilfe einer URI realisiert werden kann. In Abb. 6 wird das Statement für folgende Aussage veranschaulicht: „Das Sachbuch hat einen Autor Max Mustermann.“ „Sachbuch“ ist die Ressource (Subjekt), das „hat#autor“ ist die Property (Prädikat) und das Literal, in diesem Fall der String „Max Mustermann“ (Objekt). Diese Graphen enthalten drei Kernelemente. Knoten, die ein Subjekt darstellen, den dazugehörigen Pfeil für das Prädikat und Kanten für das Objekt.

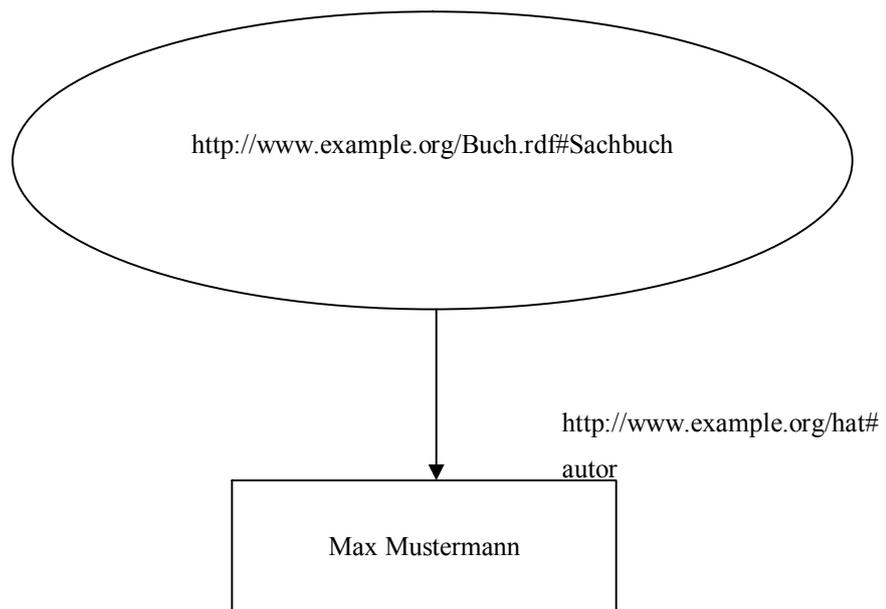


Abb. 6 Beispiel für ein Statement in RDF

Um diese graphische Darstellung jedoch auch maschinenlesbar zu machen bedarf es einer korrekten syntaktischen Darstellung. Neben der weitverbreiteten XML-Syntax existieren noch weitere mehr oder weniger offizielle Schreibweisen. Diese Kapitel bezieht sich ausschließlich auf die XML-Syntax, da diese im Gegensatz zu den anderen Sprachen auch die meiste Unterstützung findet (vgl. HITZLER 2008, S.42). Die Aussage in Abb. 6 wird folgendermaßen in der XML-Syntax ausgedrückt:

```
<?xml version="1.0?">
<rdf:RDF xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-ns#
    xmlns=http://www.example.org/Buch#>
  <Buch rdf:ID="Sachbuch">
    <autor>Max Mustermann</autor>
  </Buch>
</rdf:RDF>
```

Die erste Zeile definiert einen XML-Bereich. Zeile zwei definiert dieses Dokument als RDF-Dokument und erstellt den Namensraum „rdf“ mit der dazugehörigen URI. Darüberhinaus wird ein weiterer Namensraum erstellt, nämlich „<http://www.example.org/Buch#>“. Sämtliche Elemente, die kein Präfix haben, werden nun auch diesem Namensraum zugeordnet. Die dritte Zeile enthält das Element „Buch“, welches zu eben jenem Namensraum gehört und das Element „rdf:ID“, was aus dem RDF-Vokabular stammt und nur in Verwendung mit dem Namen der Ressource zu verwenden ist (keine URI vorhanden).

Eine Verwendung der kompletten URI ist durch den Einsatz von „rdf:about“ möglich (vgl. POWERS 2003, S. 44ff). Bistlang kann also angenommen werden, dass die Ressource „Sachbuch“ eine Instanz der Klasse „Buch“ ist, die wiederum im Namensraum „<http://www.example.org/Buch#>“ näher definiert wurde. Die vierte Zeile enthält das Property „autor“ der Klasse Sachbuch mit dem Wert „Max Mustermann“ (Literal).

Um eine Strukturierung der Werte zu erreichen bedarf es u.U. eines sog. leeren Knotens. Es handelt sich dabei um eine leere Ressource, die dazu dient mehrere einzelne Werte einer benannten Ressource zuzuordnen (vgl. ECKSTEIN, ECKSTEIN 2004, S. 240).

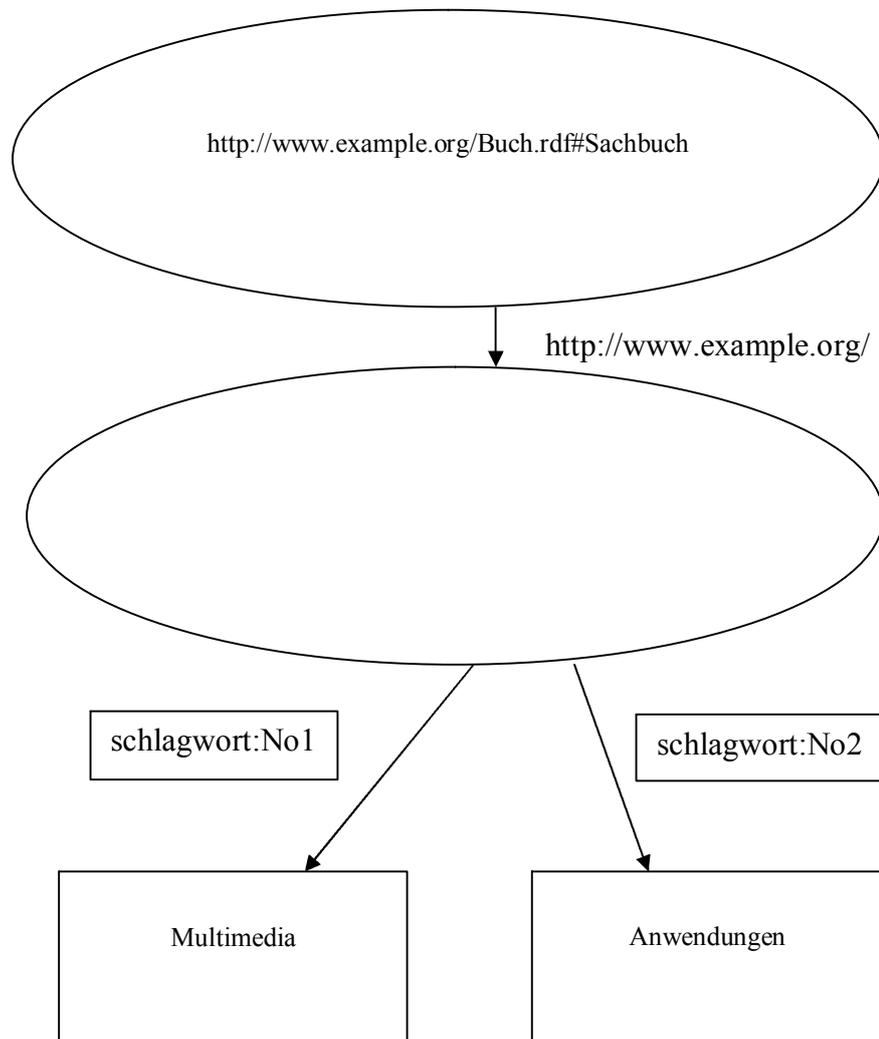


Abb. 7 Beispiel für ein Statement mit leerem Knoten in RDF

Darüberhinaus enthält das RDF-Vokabular nützliche Bestandteile für eine weitere Strukturierung. Zum einen existiert die Möglichkeit durch die Nutzung von „rdf:value“ bei einer Angabe, die aus mehreren Elementen besteht, in ein Haupt- und beschreibende Elemente zu unterteilen und damit eine Prioritätenreihenfolge herzustellen. Ein mögliches Beispiel wäre eine Angabe über eine bestimmte Dauer eines Prozesses, z.B. „18 Stunden.“ Mit

„rdf:value“ könnte die „18“ als Hauptelement beschrieben werden und „Stunden“ als Element, welches „18“ genauer beschreibt (vgl. MANOLA, MILLER 2004). Zum anderen bietet die Angabe „rdf:type“ an, einer Ressource sämtliche Eigenschaften einer bestimmten Klasse beizuordnen und somit die Ressource zu einer Instanz der Klasse zu machen (vgl. ECKSTEIN, ECKSTEIN 2004, S. 247). Später folgt ein Beispiel im Bereich RDF-Schema.

Hat man es mit mehreren Elementen zu tun, gibt es sog. Containerelemente. Diese Container gruppieren die verwandten Elemente. Es gibt mehrere Möglichkeiten: „rdf:bag“ ermöglicht eine unsortierte Listenbildung, „rdf:Seq“ hingegen eine sortierte Listenstellung (z.B. alphabetisch). Außerdem gibt es „rdf:Alt“. Es erstellt eine Liste alternativer Werte (vgl. REFNES DATA 2008 – RDF CONTAINERS). Die Container bieten die Möglichkeit offene Elementlisten zu erstellen, möchte man aber eine geschlossene Elementliste erzeugen, benötigt man sog. Collections³². Sie werden mit dem Befehl „rdf:parseType=‘Collection‘“ und der dazugehörigen Liste an Begriffselementen erzeugt (vgl. MANOLA, MILLER 2004).

Der nächste Schritt ist die Bildung eines Vokabulars für die ein darzustellendes Interessengebiet. Konkret bedeutet das, dass zu die individuellen Elemente innerhalb eines RDF-Dokuments in umfangreichere Zusammenhänge gesetzt werden müssen. Das Resource Description Schema (RDFS) ermöglicht die Konstruktion solcher Zusammenhänge und damit auch die Gestaltung einfacher (leichtgewichtiger) Ontologien. RDFS, welches ein Bestandteil von RDF ist, kann durch Einsatz von terminologischem Wissen Begriffe eines Vokabulars spezifizieren (vgl. HITZLER 2008, S. 66f). Die relevantesten Konzepte in RDFS sind die Beziehungen zwischen Klassen und Unterklassen, sowie zwischen Eigenschaften und Untereigenschaften. Unterklassen („rdfs:subClassOf“) stellen Teilmengen von Klassen („rdfs:Class“) dar. Das gleiche gilt für

³² Engl. für Sammlungen.

Untereigenschaften („`rdfs:subPropertyOf`“) und Eigenschaften („`rdfs:Property`“). Ressourcen können mit „`rdf:type`“ als Klasse definiert werden, ebenso kann man mit „`rdf:type`“ Property definieren. RDFS enthält drei zentrale Klassen: „`rdfs:Ressource`“, „`rdf:Property`“ und „`rdfs:Class`“. Dies bedeutet, dass jede Ressource eine Instanz von „`rdfs:Ressource`“ ist. Das gleich gilt für „`rdf:Property`“ und „`rdfs:Class`“. Zudem existieren fünf zentrale Eigenschaften. „`rdfs:subClassOf`“ und „`rdfs:subPropertyOf`“ sind genauso, wie „`rdf:type`“ bereits bekannt. Außerdem existieren „`rdfs:seeAlso`“, für zusätzliche Informationen über eine Ressource in nicht unbedingt für Maschinen interpretierbaren Text und „`rdfs:isDefinedBy`“ als Untereigenschaft (vgl. ECKSTEIN, ECKSTEIN 2004, S. 262ff).

Hier ist nun eine Übersicht über die möglichen RDFS Klassen und RDFS Eigenschaften. Mit diesen kann man einfache Beziehungsgeflechte formen, ohne überdimensionierte Ontologie zu verwenden, die für einfache Wissensgebiete unpassend wären (vgl. ISELE 2005, S. 44f).

Klasse	Beschreibung
rdfs:Resource	Definiert die Klasse der Ressourcen. Das zu bestimmende, abstrahierte Web-Objekt.
rdfs:Class	Definiert die Klasse als solche.
rdfs:Datatype	Definiert die Klasse der Datentypen bzw. des Datenformats.
rdfs:Literal	Definiert die Klasse der Literale (Zeichenkette). Wert eines Tripels. Nummer oder Textstring
rdf:XMLLiteral	Spezifiziert Literale und Datentypen mit schon erstellten XML-Werten (Literalen).
rdf:Property	Definiert die Klasse der Eigenschaften und ist eine Instanz von Klassen.
rdf:Statement	Definiert die Klasse der Statements, also Aussagen.
rdfs:Container	Definiert die Klasse der Container. Container sind Auflistungen mehrerer Elemente.
rdfs:ContainerMembershipProperty	Definiert die Klasse der Container Eigenschaften einer zuvor bestimmten Gruppe von Containern
rdf:Bag	Spezifiziert Container. Ermöglicht eine ungeordnete Auflistung von gleichwertigen Elementen.
rdf:Seq	Spezifiziert Container. Ermöglicht eine geordnete, hierarchische Auflistung von Elementen.
rdf:Alt	Spezifiziert Container. Ermöglicht eine Auflistung von Alternativen.
rdf>List	Definiert die Klasse der Listen und ist eine Instanz von Klassen. Ermöglicht eine Listenerstellung.

Tab. 2 RDFS Klassen (vgl. ISELE 2005, S. 44)

Eigenschaft	Beschreibung	Gehört zu	Wertebereich
rdf:type	Typ einer Klasse. Kategorie.	rdfs:Resource	rdfs:Class
rdfs:subClassOf	Unterklasse einer höheren Klasse.	rdfs:Class	rdfs:Class
rdfs:subPropertyOf	Untereigenschaft einer höheren Eigenschaft.	rdf:Property	rdf:Property
rdfs:domain	Domäne der Subjekte, auf die sich die Eigenschaft bezieht.	rdf:Property	rdfs:Class
rdfs:range	Definierter Wertebereich der Eigenschaft.	rdf:Property	rdfs:Class
rdfs:label	Thema	rdfs:Resource	rdfs:Literal
rdfs:comment	Beschreibung der Ressource.	rdfs:Resource	rdfs:Literal
rdfs:member	Teil einer Klasse.	rdfs:Resource	rdfs:Resource
rdf:first	Erstes Element einer Liste.	rdf:List	rdfs:Resource
rdf:rest	Nachfolgende Elemente einer Liste.	rdf:List	rdf:List
rdfs:seeAlso	Weitere Informationen zu einer Ressource.	rdfs:Resource	rdfs:Resource
rdfs:isDefinedBy	Definition der Ressource.	rdfs:Resource	rdfs:Resource
rdf:value	Eigenschaftsbestimmung eines Wertes.	rdfs:Resource	rdfs:Resource
rdf:subject	Subjekt des RDF-Statements.	rdf:Statement	rdfs:Resource
rdf:predicate	Prädikat des RDF-Statements.	rdf:Statement	rdfs:Resource
rdf:object	Objekt des RDF-Statements.	rdf:Statement	rdfs:Resource

Tab. 3 RDFS Eigenschaften (vgl. ISELE 2005, S. 45)

Es folgt nun eine Abbildung eines RDF-Graphen, in denen auch RDFS-Elemente (leichtgewichtige Ontologie) eingebettet sind und zur Veranschaulichung dienen sollen.

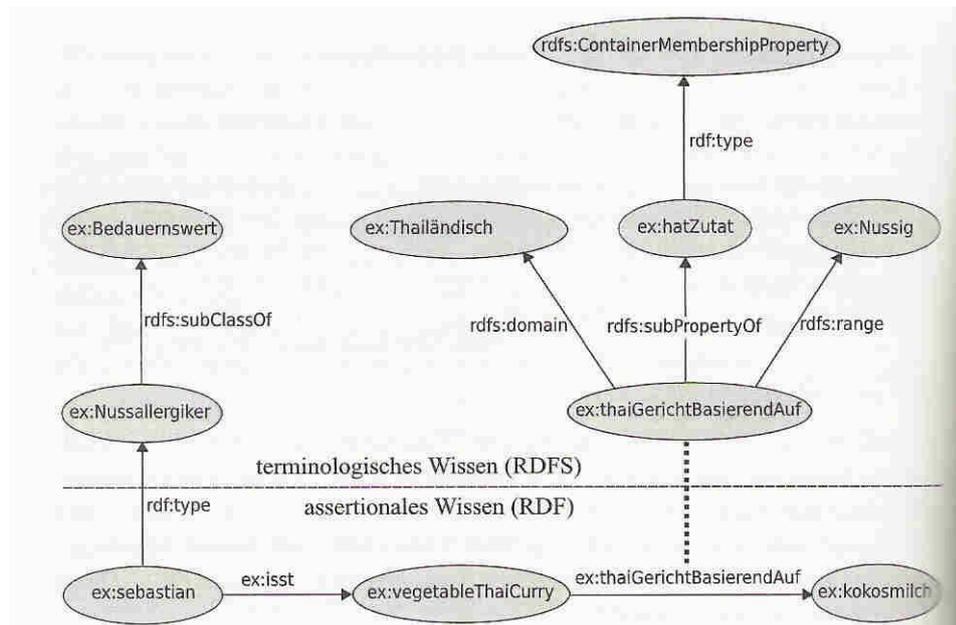


Abb. 8 RDF-Graph mit RDFS-Elementen (vgl. HITZLER 2008, S. 84)

Es ist ersichtlich, dass der untere einfache Graph, der das „assertionale Wissen³³“ beinhaltet, durch das „terminologische Wissen“ ergänzt wird und dementsprechend in Beziehung gesetzt wird. Eine erweiterte Semantik ist der Fall.

3.4 OWL

Die Web Ontology Language (OWL) spielt eine essentielle Rolle für das Semantic Web und diese Bachelorthesis. Im Gegensatz zu den vorherigen Metadatensprachen ermöglicht OWL komplexe Ontologien zu erstellen, die z.B. nicht nur Ergebnisse aus dem vorhandenen Wissen abrufen, sondern diese auch aus einzelnen Bestandteilen der Ontologie selbst konstruieren können. Dies geschieht durch die formale Semantik, die diesen Inferenzmechanismus ermöglicht (vgl. HÜSEMANN 2005, S. 58). OWL entstand aus den zwei unterschiedlichen Produkten DAML (DARPA Agent Markup Language - amerikanische Version) und OIL (Ontology Inference

³³ Faktisches Wissen.

Layer - europäische Version). Um jedoch eine Einheitlichkeit und damit eine Referenz für den Bereich der Web-Ontologien zu schaffen, entwickelte eine Arbeitsgruppe des W3C OWL und löste damit die „Randerscheinung“ DAML&OIL ab (vgl. ISELE 2005, S. 51). Da der Umfang einer Ontologie durchaus variabel sein kann, existieren drei Abstufungen von OWL, auf die nun im einzelnen kurz eingegangen wird.

OWL Full

OWL Full ermöglicht die Konstruktion komplexester, schwergewichtiger Ontologien. Es enthält sämtliche Konstrukte von RDF(S) (vgl. HÜSEMANN 2005, S.53). Ein valides OWL Full-Dokument ist daher auch ein valides RDF(S)-Dokument. Aufgrund der Ausdruckstärke und des hohen Entwicklungspotentials sind Schlussfolgerungen nicht immer möglich. Hier erfolgt eine Vermischung der Entitäten und es kommt zu langen Berechnungszeiten. Außerdem ist eine Unterstützung von OWL Full-Dokumenten durch aktuelle Tools nicht garantierbar (vgl. HITZLER 2008, S. 127).

OWL DL³⁴

OWL DL bietet ein geringeres Entwicklungspotential für Ontologien. Des Weiteren enthält OWL DL nicht alle RDF(S) Konstrukte. Schlussfolgerungen sind mit einer höheren Wahrscheinlichkeit erreichbar, als dies bei OWL Full der Fall ist. Dies kann z.B. durch die Restriktion von einzelnen Elementen geschehen (vgl. POWERS 2003, S. 233). Auch die Unterstützung durch aktuelle Tools ist bei OWL DL weitestgehend gewährleistet (vgl. HITZLER 2008, S. 127). Jedes valide und legale OWL DL-Dokument ist auch ein valides und legales OWL Full-Dokument.

³⁴ Description Logic (engl.) = Beschreibungslogik.

OWL Lite

Es ermöglicht die Konstruktion einfacher Ontologien (z.B. Taxonomien), bei der eine Schlussfolgerung generell immer möglich ist (die Kardinalität ist immer entweder 0 oder 1) (vgl. POWERS 2003, S. 233). Der Zeitaufwand einer Schlussfolgerung ist daher deutlich geringer, als bei OWL Full bzw. OWL DL. OWL Lite eignet sich durch geringere Ausdrucksstärke hauptsächlich für Privatanwender. Des Weiteren ist jedes valide und legale OWL Lite-Dokument auch ein valides und legales OWL DL-Dokument.

Ein OWL-Dokument beginnt, wie schon XML- und RDF(S)-Dokumente mit einem Kopfbereich, welcher die nötigen Namensräume enthält.

Anschließend erfolgt eine optionale Angabe zur Ontologie („owl:Ontology“).

```
<rdf:RDF
xmlns="http://www.semanticweb-grundlagen.de/beispielontologie#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:owl="http://www.w3.org/2002/07/owl#">
...
<owl:Ontology rdf:about="Beispiel">
<rdfs:comment>Beispielontologie</rdfs:comment>
...
</rdf:RDF>
```

Die zentralen Sprachkonstrukte von OWL sind Klassen, Propertys³⁵ und Individuals.

Klassen in OWL sind vergleichbar mit Klassen in RDFS und werden mit „owl:Class“ gebildet. Von vornherein existieren die Klassen „owl:Thing“ als übergeordnete Klasse, aller Klassen und die leere Klasse „owl:Nothing“ (vgl. SMITH, WELTY ET AL. 2006). OWL hat ein umfangreiches Repertoire zur Klassenkonstruktion.

³⁵ In der deutschen Fachliteratur wird auch der Begriff „Rollen“ verwendet (vgl. HITZLER 2008).

Diese unterteilen sich in vier Kategorien:

Werteinschränkungen (owl:allValuesFrom, owl:someValuesFrom, owl:hasValue), Kardinalitätseinschränkungen (owl:cardinality, owl:minCardinality, owl:maxCardinality), Mengenkonstruktoren (owl:intersectionOf, owl:unionOf, owl:complementOf) und Axiome (owl:equivalentClass, owl:disjointWith). Die Kategorie Werteinschränkungen nimmt Einfluss auf die Wertebereiche der verwendeten Eigenschaften und reguliert diese je nach Vorgabe. Kardinalitätseinschränkungen regulieren die Mächtigkeit der verwendeten Eigenschaften. Mengenkonstruktoren erlauben Klassen in verschiedene Mengenbeziehung zu setzen, z.B. Bildung einer Schnittmenge oder einer Vereinigungsmenge aus zwei Klassen. Axiome ermöglichen dagegen eine genauere Beschreibung der Instanzverhältnisse von Klassen. Hier können Klassen als semantisch gleich (äquivalent), bzw. als semantisch ungleich (disjunkt) deklariert werden. Dies wird dann auf die Instanzen der Klassen übertragen (vgl. HÜSEMANN 2005, S. 55).

Die Propertys (oder auch Eigenschaften) werden in OWL in zwei verschiedene Typen unterteilt. Der „owl:ObjectProperty“ steht für eine Beziehung zwischen Instanzen zweier Klassen, während „owl:DataTypeProperty“ Beziehungen zwischen Instanzen von Klassen, sowie von (u.U. getypten) Literalen. Einschränkungen der Gültigkeits- und Wertebereiche sind über „rdfs:domain“ und rdfs:range“ realisierbar (vgl. HÜSEMANN 2005, S. 56) . Folgende OWL-Konstrukte zur Property-Beschreibung sind in drei Kategorien vorhanden: Relationen zu anderen Eigenschaften (owl:inverseOf, owl:equivalentProperty), lokale Eigenschaften (owl:TransitiveProperty, owl:SymmetricProperty), globale Kardinalität (owl:FunctionalProperty, owl:InverseFunctionalProperty). Relationen zu anderen Eigenschaften bestimmen, ob die Beziehungen invers sind, also zwei Eigenschaften für ein Tripel verwendet werden können und eine Umkehr des Tripels ermöglicht wird oder äquivalent, also eine Eigenschaftsgleichheit vorliegt. Bei lokalen Eigenschaften handelt es sich um die Verwendung einzelner Eigenschaften. Es existieren transitive

Eigenschaften, d.h. ein transitives Tripel entsteht, wenn $X a Y$ und $Y a Z$, auch $X a Z$ ist. Hierbei steht „a“ für die Eigenschaft. Außerdem gibt es symmetrische Eigenschaften, die eine Umkehrung des Aussagentripels zulässt, ohne den Sinn zu verändern. Globale Kardinalitäten regulieren zum einen, durch einfache funktionale Eigenschaften eine Wertzuweisung zu einer Eigenschaft ($X a Y$ und $X a Z$, dann folgt daraus $Y=Z$). Darüber hinaus gibt es die invers funktionale Eigenschaft ($X a Z$ und $Y a Z$, dann folgt daraus $Y=Z$) (vgl. HÜSEMANN 2005, S.57). Individuen sind RDF-Instanzen von Klassen und sind vergleichbar mit Objekten in RDFS. In OWL stehen zur Bezeichnung von Literalen neben „xsd:string“ und „xsd:integer“ auch „rdfs:literal“ zur Verfügung (HITZLER 2008, S. 131). Individuals können untereinander in Relation gesetzt werden und zwar mit den Konstrukten „owl:sameAs“, „owl:differentFrom“ und „owl:AllDifferent“. Hier können Instanzen als äquivalent, verschieden und paarweise verschieden, wenn es sich um eine Menge von Instanzen handelt, deklariert werden (vgl. HÜSEMANN 2005, S. 58).

3.5 MPEG-7

Dieses Unterkapitel dient als Exkurs in eine bereits vorhandene Möglichkeit der Verwaltung von multimedialen Daten anhand eines Metadatenstandards. Der MPEG-7³⁶ Standard ist anders, als z.B. MPEG-4 kein Kompressionsstandard und enthält keine eigentlichen multimedialen Daten. Der Standard dient ausschließlich zur Beschreibung von Daten (MANJUNATH 2002, S. 8). Dabei kann es sich um Bilder, Videos, dreidimensionale Objekte, Audiodokumente oder auch kombinierte Daten handeln (KOSCH 2004, S. 25). Ziel der Entwicklung von MPEG-7 war die zunehmende Vielfalt von multimedialen Daten besser zu archivieren und vor allem wiederauffindbar zu machen. Darüber hinaus sollte dieser Standard für eine gewisse Objektivität und Ordnung bei der inhaltlichen

³⁶ MPEG steht für Moving Pictures Expert Group

Erschließung von multimedialen Daten sorgen. Außerdem sollte es ermöglicht werden, eine komplett manuelle Annotation obsolet zu machen und auf (semi-)automatische Erschließungsverfahren mit Hilfe von MPEG-7 umzusteigen (MANJUNATH 2002, S. 7). Man kann daher drei Ziele formulieren:

- Beschreibung von multimedialem Inhalt,
- eine flexible Verwaltung der Metadaten und
- eine globale Verwaltung der Ressourcen (NACK, LINDSAY 1999, S. 73).

Vor allem die globale Verwaltung der Ressourcen und die dazugehörige Unabhängigkeit von bestimmten Systemen trägt zur Flexibilität dieses Standards bei und ist daher ein großer Vorteil bei der Verwendung.

MPEG-7 unterteilt sich in vier wesentliche Bestandteile:

Descriptors (D)

Ein Descriptor dient zur Merkmalsbeschreibung von bestimmten Teilen eines multimedialen Dokuments. Dabei kann es sich um sog. Low-Level-Descriptors handeln (Farbe, Lautstärke, Titelinformationen) oder High-Level-Descriptors, die bereits komplexe semantische Informationen enthalten. MPEG-7 enthält bereits eine Reihe von Descriptors, die Syntax und Semantik definieren und repräsentieren können. Es ist möglich individuelle Descriptors zu erstellen.

Description Schemes (DS)

Description Schemes dienen zur Erstellung von Relationen zwischen verschiedenen Descriptoren, aber auch zwischen unterschiedlichen Descriptor Schemes. Mit Hilfe eines DS können semantische Strukturen und Hierarchien erstellt werden, ähnlich wie es bei RDF(S) der Fall ist.

Description Definition Language (DDL)

Zur Erstellung neuer Descriptors und neuer Description Schemes bedarf es einer Definitionssprache. Die Description Definition Language ist eine solche Sprache. Ihre syntaktische Basis ist eine leicht modifizierte Fassung von XML Schema (vgl. STAMOU, KOLLIAS 2005, S. 15).

Systems Tools

Systems Tools übernehmen hauptsächlich die Verwaltung der erstellten Metadaten, wie z.B. die Archivierung, Transport und Synchronisation. Außerdem dienen sie dem Schutz des geistigen Eigentums (MANJUNATH 2002, S. 14f)

Wie bereits in den Unterkapiteln zuvor erwähnt, ist XML-Schema in der Gestaltung von Ontologien, im Gegensatz zu OWL z.B., deutlich im Nachteil. Die DDL, die auf XML-Schema aufbaut verfügt nicht über das komplexe Strukturangebot. Deshalb soll u.a. untersucht werden, ob es Sinn ergibt, eine Multimedia-Ontologie für Bild und Videodateien zu entwickeln, die sich die umfangreichen Strukturen und Mechanismen zu Nutze machen.

4. Inhaltsextraktion

Um Dokumente, ganz gleich ob alphanumerisch oder multimedial, für Such- und Abfrageprozesse zugänglich zu machen, müssen sie zunächst inhaltlich erschlossen werden. Dieses Kapitel fasst einige Methoden zur inhaltlichen Erschließung von alphanumerischen Dokumenten wie Texten, Büchern und Webseiten sowie von multimedialen Dokumenten zusammen. Vor allem stehen Foto- sowie Videodokumente im Fokus. Unterteilt wird die Methodendarstellung in manuelle und (semi-)automatische Erschließungsverfahren. Es sollen weiterhin Vor- und Nachteile der unterschiedlichen Verfahren präsentiert werden.

4.1 Inhaltsextraktion alphanumerischer Dokumente

Neben der formalen Erschließung ist die inhaltliche Erschließung ein Aufgabengebiet, welches zu einem großen Teil Bibliotheken, Dokumentationsstellen und Archive übernehmen. Eine inhaltliche Erschließung von Textdokumenten meint eine Verschlagwortung, Klassifikation und/oder Erstellen eines Abstracts. Im Gegensatz zur formalen Erschließung ist die inhaltliche durchaus komplexer und wird daher vornehmlich von Fachspezialisten (Fachreferenten) im Bibliotheks- und Dokumentationsbereich ausgeübt (WIKIPEDIA 2008 – SACHERSCHLIEßUNG). Eine Verschlagwortung bzw. Klassifikation erfolgt durch ein vorgegebenen Bestand an Schlagworten. Dies ist ein wesentlicher Vorteil, da es sich z.B. um allgemeingültige Thesauri oder Systematiken handelt, die über die Grenzen einzelner Bibliotheken und Dokumentationseinrichtungen anerkannt sind (vgl. BLANKEN, BLOK ET AL. 2007, S. 98). Im deutschen Bibliothekswesen sind die „Regeln für den Schlagwortkatalog“ (RSWK) maßgebend. Im weniger professionellen Umfeld, z.B. bei Privatanwendern ist die Vergabe von freien Termini

verbreitet. Diese Schlagwörter sind schnell und unkompliziert zu vergeben. Jedoch haben sie den Nachteil, sehr subjektiv zu sein. Das verwendete Vokabular ist nicht standardisiert, so dass Verwechslungen und Missverständnisse möglich sind (vgl. UMLAUF 2000).

Die automatische Inhaltsextraktion beinhaltet meist mehrere Verfahren, um ein akzeptables Ergebnis an Metadaten zu erhalten. Dies geschieht mit Hilfe von Algorithmen³⁷, die dabei verschiedenste Aufgaben zu lösen haben. Als erstes bedarf es einer genauen Erkennung, um welche Sprache es sich bei dem Dokument handelt. Daraufhin werden die Stoppwörter, wie z.B. Artikel, Präpositionen, Konjunktionen entfernt, da sie mit dem Kontext nicht direkt in Verbindung zu bringen sind. Anschließend könnte ein sog. Stemmingverfahren, die verwendeten restlichen Wörter auf ihren Wortstamm reduzieren (vgl. BLANKEN, BLOK ET AL. 2007, S. 98). Anschließend kann durch ein, für den Arbeitszweck individuelles, Wörterbuch ein Abgleich erfolgen um damit die nötigen Metadaten kenntlich zu machen (vgl. BERTRAM 2005, S. 106). Ein weiterer Ansatz dazu wäre ein statistisches Verfahren zur Bestimmung von Termgewichten. Es existieren zwei Formeln zur Berechnung der Termfrequenz für ein Dokument.

³⁷ "Ein Algorithmus ist eine Verarbeitungsvorschrift, die so präzise formuliert ist, daß sie von einem mechanisch oder elektronisch arbeitenden Gerät durchgeführt werden kann." (BERTRAM 2005, S. 104)

$$TF(td) = \frac{\text{Häufigkeit eines Wortes im Dokument}}{\text{Anzahl aller Wörter des Dokuments}}$$

t = Term

d = Dokument

TF = Termfrequenz

(vgl. SCHULZ 2008)

Je mehr Wörter ein Textdokument enthält, desto mehr Nullnachkommastellen enthält die Angabe der Termfrequenz. Aus diesem Grund existiert eine modifizierte Version der Formel.

$$TF(td) = \frac{\log_2(\text{Häufigkeit von t in d} + 1)}{\log_2 \text{Gesamtanzahl der Wörter im Dokument}}$$

t = Term

d = Dokument

TF = Termfrequenz

(vgl. SCHULZ 2008)

Sind für die vorhandenen Terme die jeweiligen Frequenzen errechnet worden, kann eine Rangliste einen Überblick geben. Je nachdem wie viele Schlagwörter gebraucht werden, können dann eine Reihe der besserplatzierten Terme als Metadaten für das entsprechende Dokument verwendet werden.

Lexikonbasierte und statistische Verfahren können auch kombiniert werden, um die Ergebnisqualität zu steigern (vgl. UMLAUF 2000). In der Praxis werden diese automatischen Erschließungsmethoden in Pressebereichen mit hoher

Dokumentmasse eingesetzt, um die Dokumentation zu beschleunigen und Kosten zu sparen. Jedoch bedarf es weiterhin z.T. intellektueller Kontrolle, um Fehlerkorrektur zu ermöglichen und damit eine hohe Ergebnisqualität auch tatsächlich zu ermöglichen (BERTRAM 2005, S. 113).

4.2 Inhaltsextraktion multimedialer Dokumente

Ähnlich wie bei alphanumerischen Dokumenten bestehen Möglichkeiten manuell inhaltlich zu erschließen. Im Unterschied zu Büchern, Zeitschriftenartikeln etc. ist dies weniger eine Aufgabe für Bibliothekare, sondern eher für Archivare und seit der Entstehung von multimedialen Plattformen, wie z.B. MyVideo oder YouTube auch immer mehr für Privatanwendern, die selbst Inhalte veröffentlichen. Solche Plattformen bieten meist auch eine freie Verschlagwortung (Folksonomie) an, die wie bereits im Unterkapitel zuvor erwähnt, durchaus Nachteile hat. Diese treten besonders zum Vorschein, wenn es sich um große Mengen von Dokumenten handelt. Aus diesem Grunde sollen vor allem (semi-)automatische Erschließungsverfahren für multimediale Dokumente in diesem Unterkapitel thematisiert werden, die die Arbeit von Archivaren bzw. Dokumentaren nicht ersetzen, sondern eher ergänzen sollen (vgl. HERMES, MIENE ET AL. 2003, S. 101).

4.2.1 Inhaltsextraktion von Bilddateien

Inhaltsextraktion von Bildern erfolgt hauptsächlich anhand der Identifikation von drei Kernfaktoren. Es handelt sich dabei um Farben, Texturen und Konturen, die analysiert werden (vgl. MIENE, HERMES ET AL. 2002, S. 16). Die Farbanalyse erfolgt mit Hilfe einer Farbsegmentierung. Das verwendete Bilddokument soll in gleichartige Farbbereiche unterteilt werden. Diese Unterteilung in Bildregionen hat bestimmte Maße zur Anforderung. Zu kleine, gleichfarbige Regionen verschmelzen mit der jeweiligen Nachbarregion. Ein weiteres Kriterium für autonome Regionen eines Bildes sind eine möglichs geringe Farbabweichung und lange gemeinschaftliche Grenzen. Die Farbdarstellung erfolgt im HLS-Farbsystem und nicht im RGB-Farbschema, da

es der menschlichen Farbwahrnehmung eher entspricht. Im HLS-System werden Farben ihre Farbwerte (H), Helligkeit (L) und ihre Sättigung (S) präsentiert. Ist die Segmentierung abgeschlossen, erfolgt eine Zusammenfassung und Benennung der Farben anhand der HLS-Werte (vgl. HERMES, MIENE ET AL. 2003, S. 101f). Eine Segmentierung ist auch bei der Texturanalyse der erste Schritt. Hier werden jedoch nicht Farben segmentiert, sondern die im Bild befindlichen Oberflächenstrukturen. Es werden Texturflächen anhand ihrer Größe und Position innerhalb einer bestimmten Bildregion beschrieben. Die Segmentierungsverfahren unterscheidet man in ein bereichsbasiertes Textursegmentierungsverfahren (vollständige Segmentierung) und ein kantenbasiertes Segmentierungsverfahren (unvollständige, auf Bildregionen basierende Segmentierung). Nach Abschluss der Segmentierung werden Proben der Texturen entnommen, statistisch bezüglich ihrer visuellen Merkmale (Kontrast, Grobheit, Regelmäßigkeit etc.) analysiert und dargestellt. Anhand dieser kann eine Anwendungsdomäne erkannt werden und Objekte aus den jeweiligen Klassen können mit den untersuchten Texturen abgeglichen werden (vgl. MIENE, HERMES ET AL. 2002, S. 18). Eine Konturanalyse beschreibt die Umrisse des analysierten Bildes. Eine Ansammlung von Bildpunkten (Kette) ergibt ein Konstrukt, welches mit einer Reihe von geometrischen Figuren abgeglichen werden und dann je nach Ähnlichkeit zugeordnet werden kann. Eine Kombination der Analysen aller drei Kernfaktoren ermöglicht eine Extraktion von Informationen in Form von Metadaten (vgl. HERMES, MIENE ET AL. 2003, S. 102f). Die folgende Abbildung zeigt, wie anhand dieser Analyseverfahren ein konkretes Objekt innerhalb eines Fotos identifiziert wird.

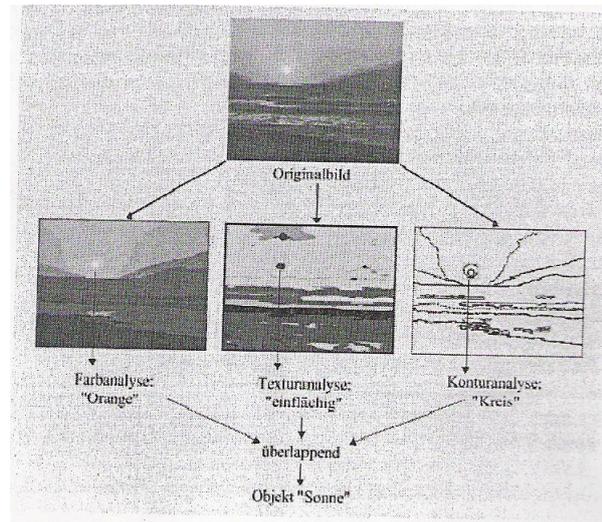


Abb. 9 Inhaltsextraktion eines Fotos (vgl. HERMES, MIENE ET AL. 2003, S. 102)

Eine Objekterkennung kann auf mehreren Wegen erfolgen (BLANKEN, BLOK ET AL. 2007, S. 162ff). Der einfache Abgleich mit Beispielobjekten einer bestimmten Klasse wurde bereits erwähnt. Hier werden Pixelwerte mit einander verglichen, um das passende Objekt zu finden, um es dem analysierten Bild zuordnen zu können. Die „Eigenobjects-Erkennung“ basiert ebenfalls auf einem Abgleich von visuellen Daten mit Daten in einer Datenbank. Diese algorithmische Methode berechnet ein Vektorgewicht für einzelne Bildelemente und vergleicht diese mit den Daten innerhalb der Datenbank. Entspricht das Ergebnis der Berechnung in einem Datensatz, dann kann eine Zuordnung erfolgen. Diese Erkennungsmethode wird häufig zur Gesichtserkennung eingesetzt. Der statistische Ansatz der Objekterkennung enthält eine Formerkennung eines Objektes mit Hilfe einer Begrenzung, durch sog. verbundene Landmarks³⁸, um z.B. mehrere ähnliche Objekte innerhalb eines Bildes differenzieren zu können. Durch einige Beispielmodelle können statistische Abgleiche erfolgen, die eine nähere Definition des Objektes zulassen.

³⁸ Hier: Abgrenzung der Form eines Objektes in einem Bild.

4.2.2 Inhaltsextraktion von Videodateien

Bei Videos handelt es sich im Detail um Bildfolgen. Da eine Verarbeitung von einzelnen Bildern innerhalb eines Videos sehr aufwendig bzw. kaum zu bewältigen ist, gibt es alternative Ansätze in der (semi-)automatischen Inhaltsextraktion von Videodateien. Generell wird ein Video in einzelne Kameraeinstellungen (Shots) unterteilt, die einzeln mit einem Schnitt beendet werden. Es erfolgt eine Berechnung von Merkmalen, die sich an einem Schwellenwert orientieren. Wird dieser überstiegen, dann ist die Grenze eines Shots erreicht. Dies funktioniert bei sog. harten Schnitten, also dort wo ein direkter Übergang sichtbar ist, sehr gut. Jedoch ist es auch möglich, wenn auch deutlich komplizierter, weiche Schnitte erkennen zu lassen. Dies sind Schnitte, die einen Spezialeffekt beim Übergang von Szenen beinhalten (Ein- und Ausblendung, Überblendung). Durch drei Verfahren (Fourier-Analyse, Grauwert-Histogramm und YUV-Farbwerten) wird dieser Vorgang realisiert. Als erstes erfolgt eine Vorverarbeitung von Merkmalen, bekannte Shotgrenzen werden dann für alle drei Verfahren dokumentiert und anhand einer Berechnung von Schwellenwerten werden abschließend in einer Liste die genauen Shotgrenzen gesetzt (vgl. MIENE, HERMES ET AL. 2002, S. 19). Die Erkennungsraten liegen bei ca. 80 Prozent für weiche Schnitte und annähernd 100 Prozent für harte Schnitte (vgl. HERMES, MIENE ET AL. 2003, S. 104). Ein Großteil der automatischen Extraktionsverfahren beschränkt sich auf die Erkennung und Kombination von Low-Level Descriptors (vgl. KOMPATSIARIS, HOBSON 2008, S. 253). Ein Ansatz zur Verwendung von semantischen Konzepten ist noch nicht wirklich ausgereift und trägt weiterhin dazu bei, dass vor allem bei Archiven und Dokumentationseinrichtungen die intellektuelle Erschließung unabdingbar ist, um eine akzeptable Qualität zu garantieren. MPEG-7 ist in diesem Bereich durchaus eine sinnvolle Verstärkung der automatischen Möglichkeiten, da es bereits eine umfangreiche Auswahl von Deskriptoren anbietet. Das sind z.B. Farbdeskriptoren, Texturdeskriptoren, wie sie ähnlich bei Inhaltsextraktion von Bildern auftauchen oder auch

Bewegungsdeskriptoren, speziell für Bildfolgen (vgl. STAMOU, KOLLIAS 2005, S. 165ff).

Eine inhaltliche Strukturierung eines Videos benötigt in der Praxis einen Zugriff auf vorhandenes Wissen. Das bedeutet, dass Videos nach einem bekannten Muster ablaufen müssen, um eine Erkennung zu ermöglichen (vgl. HERMES, MIENE ET AL. 2003, S. 104).

5. Retrieval und Suchverhalten

Nachdem die inhaltliche Erfassung der alphanumerischen und multimedialen Dokumente im vierten Kapitel im Vordergrund stand, geht es nun um das Suchen und Finden relevanter Dokumente.

Zunächst soll die Definition für den Fachbegriff „Retrieval“ gegeben und die verschiedenen Retrievalmethoden- und -modelle präsentiert werden. Analog zum vierten Kapitel wird auch in diesem Kapitel getrennt zwischen alphanumerischen und multimedialen Daten, deren Retrieval auch hier den einen oder anderen Unterschied aufweist. Der Themenkomplex „Suchverhalten“ wird hinterher differenziert dargestellt. Hier sollen die Begriffe „Searching“ und „Browsing“ näher erläutert werden. Zu guter Letzt soll auf den Begriff der „Ergebnisqualität“ eingegangen werden. Hier spielen die Begriffe „Precision“ und „Recall“ eine wichtige Rolle.

5.1 Definition, Methoden- und Modellübersicht

Der Begriff „Retrieval“ oder auch „Information Retrieval“ (IR) stammt aus dem Englischen und meint die Wiedergewinnung bzw. die Auffindung von Informationen. Das IR ist ein Teilgebiet sowohl der Informationswissenschaften und der Computerlinguistik als auch der Informatik (vgl. WIKIPEDIA 2008 – INFORMATION RETRIEVAL). Nicht zu verwechseln ist IR mit der Recherche. Die Recherche bezieht sich auf eine konkrete Suche, während das IR den gesamten Umfang der Methoden und Modelle zur Findung von Informationen enthält (vgl. KUNZ 2006, S. 27). Das Retrieval von Informationen ist generell von zwei Seiten abhängig. Zum einen von einem Autor, der ein Dokument erstellt und es dann zur Verfügung stellt und zum anderen von einem Nutzer, der Interesse an diesem Dokument hat und nach diesem sucht. Das Dokument kann in einem Retrievalsystem durch Metadaten, wie z.B. Schlagwörter oder Abstracts dargestellt werden. Der Nutzer kann dann eine Anfrage formulieren, die dann mit der

Dokumentdarstellung abgeglichen wird. Diesen Vorgang nennt man auch „Matching“ (vgl. BLANKEN, BLOK ET AL. 2007, S. 99). Anschließend wird eine Trefferliste angezeigt, die entweder Treffer enthält, wenn der Abgleich erfolgreich war oder keine Treffer enthält. Nun kann der Nutzer evtl. die Darstellung der Dokumente optimieren, was allerdings nur begrenzt möglich sein sollte, da sonst die Objektivität u.U. nicht mehr garantiert wird. Er hat auch die Möglichkeit die Suche zu verfeinern, indem er die Suchanfrage ergänzt und damit die Trefferliste einschränkt oder er entscheidet sich für eine komplett andere Anfrage, weil er z.B. mit gefundenen Resultaten nicht zufrieden ist. Dieses Szenarien werden in der folgenden Grafik zusammenfassend dargestellt.

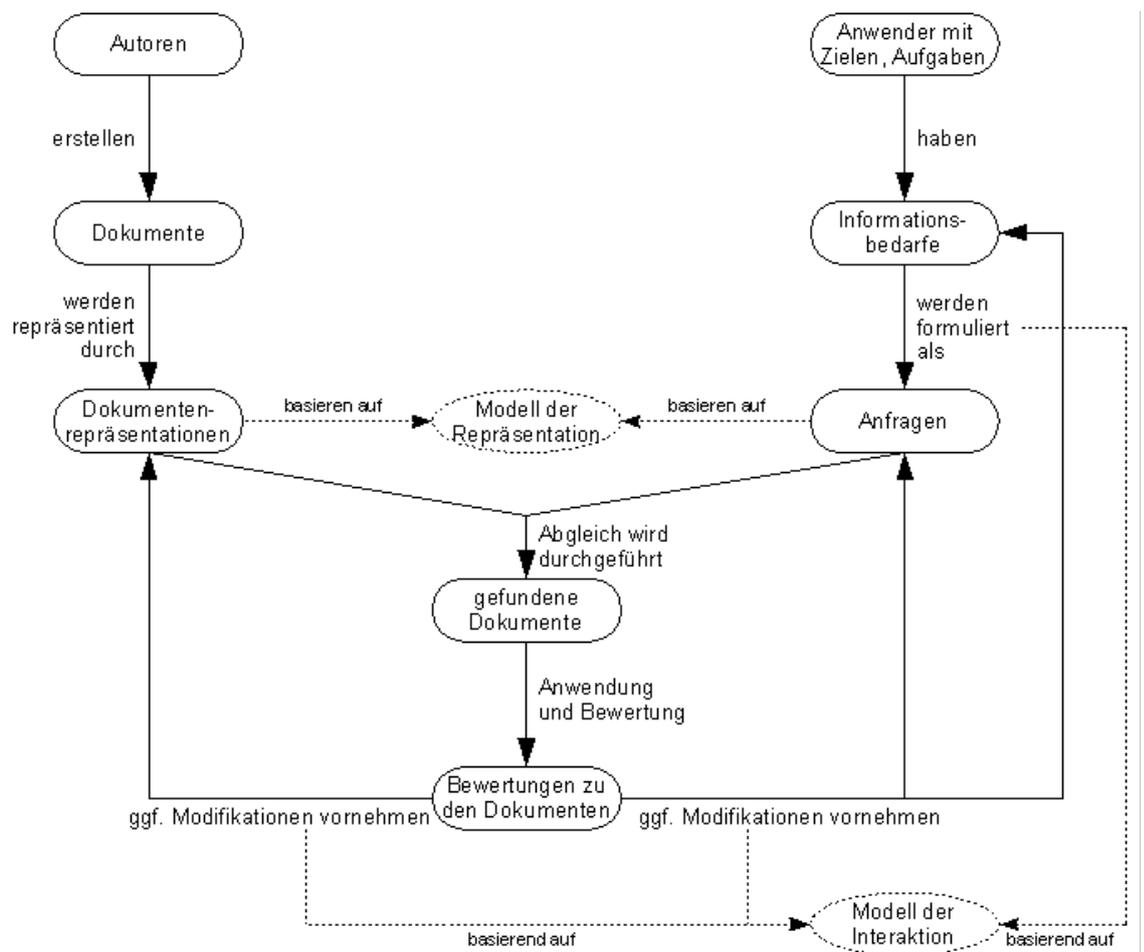


Abb. 10 Schematische Darstellung des Information Retrievals (vgl. Kuroпка 2004)

Das IR wird in mehrere verschiedene Modelle unterteilt. Diese unterscheiden sich in den Fähigkeiten der Ergebnisfindung und den Möglichkeiten eine Suchanfrage zu formulieren. Die wichtigsten Modelle sollen nun dargestellt werden.

5.1.1 Mengentheoretische Modelle

Boolesches Retrieval

Das Boolesche Retrievalmodell, benannt nach dem Mathematiker George Boole, ist vielleicht das bekannteste Retrievalmodell. Die einfache Form dieses Modells bietet die Operatoren „AND“, „OR“ und „NOT“ an. Dokumente werden indexiert und durch einfache Terme repräsentiert, allerdings ohne eine Relevanzgewichtung. Nun kann der Nutzer eine Suchanfrage mit oder ohne Operator abschicken und bekommt die Dokumente angezeigt, die den Begriff aus der Suchanfrage enthalten. Eine Kombination aus z.B. zwei Suchbegriffen mit dem Operator „AND“ ermöglicht eine Ergebnisdarstellung, deren Menge entweder gleich groß oder kleiner ist als nur mit einem der beiden Suchbegriffe. Der Operator „OR“ erzeugt eine Menge, die gleich groß oder größer ist (vgl. BLANKEN, BLOK ET AL. 2007, S. 100f). Eine Verwendung des Operators „NOT“ schränkt die Treffermenge generell ein. Aufgrund der Mengendarstellung gehört das Boolesche Retrievalmodell zu den mengentheoretischen Modellen. Ergänzt wird das Modell durch die potentielle Verwendung von Wildcards bzw. Trunkierungen. Ein Suchbegriff kann mit einer Trunkierung ausgestattet, z.B. nicht nur den Begriff in der Einzahl finden, sondern auch in der Mehrzahl oder den Begriff kombiniert mit einem anderen Begriff (vgl. STOCK 2007, S. 142f).

Beispiele für einige, häufig benutzte Trunkierungsmöglichkeiten:

Suchbegriff mit unbegrenzter Linkstrunkierung: *apfel

gefunden wird z.B.: Granatapfel, Wildapfel

Suchbegriff mit unbegrenzter Rechtstrunkierung: apfel*

gefunden wird z.B.: Apfelmilch, Apfelsaft

Suchbegriff mit unbegrenzter Links- und Rechtstrunkierung: *apfel*

gefunden wird z.B.: Granatapfelmilch, Granatapfelsaft, Wildapfelmilch,
Wildapfelsaft

Suchbegriff mit begrenzter Trunkierung in der Mitte: Me?er

gefunden wird z.B.: Meyer, Meier.

Generell ist das Modell für Laien einfach zu verstehen und anzuwenden.

Wie schon erwähnt existiert keine Relevanzgewichtung im eigentlichen Sinne in diesem Modell. Die Suche kann lediglich einen binären Wert zurückgeben.

Wenn Treffer in einer Anfrage erzielt wurden, steht das für den Wert „1“, wenn keine Treffer erzielt wurden, dann steht das für den Wert „0“. Diese Tatsache wird beim Booleschen Modell, besonders bei Fachleuten, kritisiert.

Nichtsdestotrotz finden sich Boolesche Operatoren in vielen virtuellen Bibliothekskatalogen und Datenbanken wieder, da die Nutzung schnell zu erlernen und durchaus brauchbare Ergebnisse erzielt werden können.

Erweitertes Boolesches Retrieval

Ziel des erweiterten Booleschen Retrievals ist die Schaffung einer Gewichtung der Suchterme. Im Endeffekt geht es um die Überwindung der binären Wert, die im „einfachen“ Booleschen Modell bei einer Suchanfrage zurückgegeben werden.

Ein Dokument in einer Datenbank wird mit einem Schlagwort versehen, welches ein bestimmtes Gewicht enthält. Dieses Gewicht orientiert sich entweder an der inhaltlichen Relevanz innerhalb des Dokuments oder an einem Wert, den der Nutzer definiert hat (vgl. STOCK 2007, S. 185ff). Das erweiterte Boolesche Retrieval unterteilt sich in mehrere Modellansätze, die entweder ein sog. Query³⁹-Gewicht voraussetzen, d.h. dem Suchterm wurden ein Relevanzwert zugeordnet, oder Modellansätze, bei denen die Suchterme gleichwertig sind.

MySQL-Volltextsuche

Der wohl bedeutendste relationale Datenbankstandard weltweit, MySQL, unterstützt ein Volltextretrieval. Relationale Datenbanken bestehen im Prinzip aus mehreren Tabellen (Entitäten). Diese Entitäten können indiziert werden, so dass eine Suche über mehrere Tabellen hinweg möglich ist. Die zu durchsuchenden Entitäten bilden einen FULLTEXT-Index. Mit den Befehlen MATCH() und AGAINST() kann eine Suche in dem Index erfolgen, die dann eine Ergebnisliste ausgibt. Um eine Ergebnisliste absteigend nach Relevanzwerten zu bekommen, benötigt man eine WHERE-Klausel. Darüberhinaus unterstützt MySQL das boolesche Retrieval und Retrieval mit Abfrageerweiterung. Letzteres ergänzt kurze Suchanfragen mit weiteren wichtigen Zusätzen (Extensions). Eine Stoppwortliste ist ebenfalls enthalten, jedoch nur für englische Stoppwörter. Eine Feineinstellung der Volltextsuche ist optional vorhanden. Die Länge der zu indizierenden Wörter kann damit z.B. definiert werden. Hier wird jedoch darauf hingewiesen, dass MYSQL standardmäßig bereits für die Volltextsuche optimiert ist und daher Veränderungen zu suboptimalen Ergebnissen führen können (vgl. MYSQL 2008).

³⁹ Engl. für Anfrage.

5.1.2 Vektorraum-basiertes und probabilistisches Retrieval

Das Vektorraum-Modell ist ein statistischer Ansatz beim Suchen und Finden von Dokumenten. Konkret wird die Gesamtheit der Suchterme als eigenständiges Dokument verstanden. Es kommt zu einer Erstellung eines Raummodells, dessen Dimensionen abhängig von der Anzahl der Suchterme sind (vgl. BLANKEN, BLOK ET AL. 2007, S. 104). Dann erfolgt ein Abgleich, in dem der Winkel zwischen den Vektoren berechnet wird, dieses „Dokuments“ mit den Dokumenten in der zu durchsuchenden Sammlung. Es kann nun eine Liste der Treffer erstellt werden, die einen Wahrscheinlichkeitszusammenhang mit den Suchtermen haben (vgl. BAEZA-YATES, RIBEIRO-NETO 2006, S. 27). Im Gegensatz zum Booleschen Modell gibt es durch diese Wahrscheinlichkeitsberechnung keine Restriktion auf binäre Ergebniswerte. Diese können also durchaus zwischen „0“ und „1“ liegen. Neben der Möglichkeit Trefferlisten nach Ähnlichkeit zu sortieren, existiert auch eine Relevanzsortierung nach einer Suche. Dabei handelt es sich um die Wahrscheinlichkeitsberechnung der Wichtigkeit eines Dokuments in Bezug auf eine gestellte Suchanfrage. Es handelt sich dabei um das probabilistische Modell. (vgl. STOCK 2007, S. 354f).

5.1.3 Sprachmodelle und Bayesian Retrieval

Sprachmodelle bilden Sätze in Texten als mathematisches Modell ab. Listen von Wörtern können mit der Frequenz von Wörtern in zu durchsuchenden Texten abgeglichen werden. Besonders beliebt sind statistische Methoden bei Sprachmodellen. Es werden hierbei einfache Modelle von den einzelnen Dokumenten in der Sammlung erstellt und dann mit den Anfragen verglichen. Die gefundenen Dokumente werden anschließend nach statistischer Wahrscheinlichkeit so angeordnet, wie sie am besten zum Modell der Sammlung passen (vgl. BLANKEN, BLOK ET AL. 2007, S. 115). Das Bayesian Retrieval beruht auf einem Netzwerkprinzip. Zur Darstellung werden gerichtete, azyklische Knoten für zufällige Variablen und Pfeile für

Verbindungen verwendet (vgl. BAEZA-YATES, RIBEIRO-NETO 2006, S. 48f). Azyklisch meint, dass es keine direkte Verbindung zwischen zwei bestimmten Variablen gibt. Existiert eine Verbindung, sind die Variablen voneinander, in Bezug auf ihre Wahrscheinlichkeit, abhängig. Im Umkehrschluss sind Variablen voneinander unabhängig, wenn keine Verbindung existiert. Der Vorteil eines solchen Modells ist die sichtbare Kombination von verschiedenen Variablen in Bezug auf ein bestimmtes Dokument und das unter der Einbeziehung einer bestimmten Nutzeranfrage. Der Nutzer kann durch die Verbindungen der einzelnen Variablen erkennen, dass eine Beweiskette vorliegt. Dies ermöglicht dem Nutzer ein transparentes Retrieval (vgl. BLANKEN, BLOK ET AL. 2007, S. 111ff). Das folgende Modell demonstriert, dass eine Suche nach dem Begriff „MetastaticCancer“ Verbindungen und damit u.U. auch Dokumente zu den Begriffen „SerumCalcium“ und „BrainTumor“ findet. Jedoch auch zu den Begriffen „Coma“ und „HeadAche“, obwohl diese nicht direkt mit dem Ausgangsbegriff in Verbindung stehen, aber evtl. auch von Interesse sein könnten. Diese Dokumente hätten dann allerdings definitiv kein höheres Gewicht, als die Dokumente mit den Begriffen „SerumCalcium“ und „BrainTumor“.

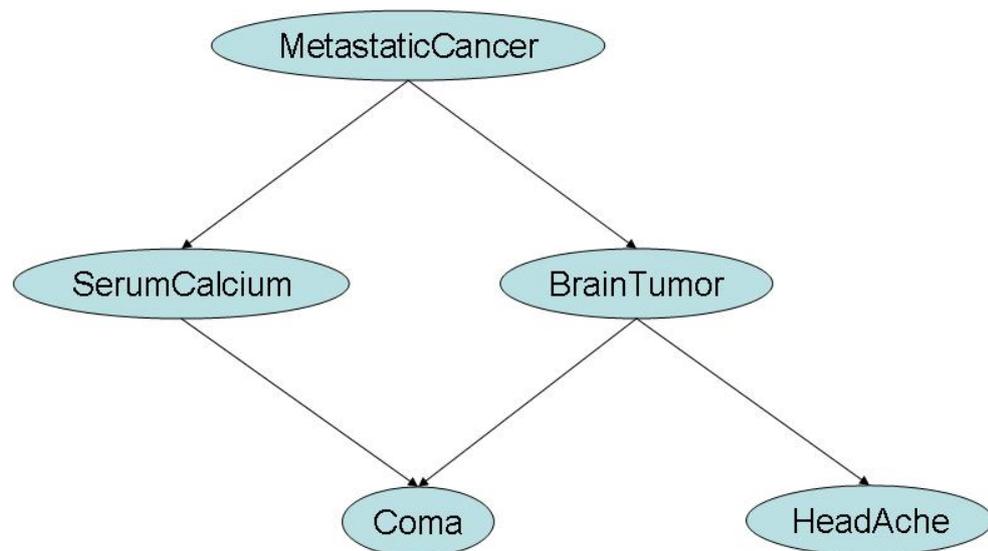


Abb. 11 Bayesian Retrieval Model (vgl. Fukushige 2004)

5.2 Retrieval von alphanumerischen Dokumenten

Das Retrieval alphanumerischer Daten läuft nach einem mehrstufigen Schema ab. Diese Schema enthält Mechanismen, wie sie z.B. im vierten Kapitel zum Thema „Inhaltsextraktion“ beschrieben wurden.

Der erste Schritt wäre das Hochladen und die dazugehörige inhaltliche Aufarbeitung des Dokuments. Anhand verschiedener Merkmale, wie z.B. dem Termgewicht, Positionen von Wörtern etc. wird ein Index erstellt, der die Suche in diesem Dokument ermöglicht. Optional kann außerdem auf semantische und strukturierte Dokumentindexe zurückgegriffen werden. Hier sollen die in den Dokumenten enthaltenen Konzepte im Retrievalsystem mit abgebildet werden. Eine vom Benutzer gestellte Anfrage wird im sog. Termprocessing mit den vorhandenen indexierten Termen der vorhandenen Dokumente abgeglichen. Dieser Prozess schließt die meist umfangreichen Möglichkeiten einer Suche, wie z.B. Boolesche Operatoren etc. mit ein. Um dem Nutzer eine möglichst nützliche Übersicht über die gefundenen Dokumente zu geben, erfolgt nach der Bearbeitung der Suchanfrage eine Relevanzberechnung (HÜRST 2005, S. 30ff).

5.3 Retrieval von multimedialen Daten

Ist die Rede von multimedialem Retrieval, so betrifft das Daten aus dem Bereichen Video, Audio, Foto, aber auch Text. Hier muss erwähnt werden, dass Retrievalsysteme für multimediale Dokumente noch in der Entwicklungsphase stecken. Erste Ansätze sind zwar vorhanden und werden in der Fachwelt diskutiert und fortentwickelt, jedoch fehlt bislang ein Durchbruch.

Ähnlich wie bei Textdokumenten kann auch das Retrieval von multimedialen Dokumenten ablaufen. Eine (manuelle) inhaltliche Erschließung durch Schlagwörter kann eine Suche dieser in Datenbanken ermöglichen, ohne dass in den Dokumenten an sich gesucht wird. Dies hat jedoch bereits bekannte

Schwächen. Es kann an Objektivität mangeln und der Aufwand bei einer großen Sammlung von Dokumenten ist meist enorm. Die Erfassung von Low-Level Descriptors, wie die Farben, Texturen etc. ist ein Ansatz eines verbesserten Retrievals (vgl. STOCK 2007, S. 512ff). Jedoch ist dem Nutzer nicht wirklich damit geholfen, Farbverlauf und Konturen eines Bildes automatisch zu extrahieren. Hier muss ein Zusammenfügen der Low-Level Descriptors zu High-Level Descriptors erfolgen, sodass relevante Konzepte eines Dokuments erfasst werden können. Dies gilt im Prinzip für alle multimedialen Formate.

Eine Suchanfrage, wie z.B. in Kapitel 5.2 erwähnt stößt bei Bildern oder Videos u.U. auf Grenzen. Versuche Musterbilder bzw. selbsterstellte Bildkonstruktionen zur vergleichenden Anfrage konnten sich bislang nicht durchsetzen (vgl. STOCK 2007, S. 515ff).

5.4 Suchverhalten

Das Suchverhalten eines Nutzes kann variieren. In der Fachliteratur wird zwischen zwei Begriffen unterschieden. Dabei handelt es sich um die Begriffe „Searching“ und „Browsing“. Vor dem Hintergrund, dass die Suche mit der vermehrten Speicherung und Archivierung von multimedialem Content mannigfaltiger geworden ist, ist eine kurze Einführung in diese beiden Begriffe von Vorteil.

5.4.1 Searching

Der Begriff „Searching“ meint im Grunde die Suche unter Verwendung konkreter Suchanfragen z.B. an eine SQL-Datenbank. Der Nutzer hat eine konkrete Vorstellung, wie das Suchergebnis aussehen könnte und welche Inhalte für ihn relevant sein könnten. Die angesprochene Mannigfaltigkeit führt zu neuen Ansätzen von Abfragesystemen. Neben der Möglichkeit der Suche nach Metadaten, spielen z.B. Suchprofile die beispielsweise von anderen

Nutzern erstellt wurden, eine große Rolle. Bereits gestellte Suchanfragen, in welcher Form auch immer, können von verschiedenen Nutzern mit verwendet werden. Suchansätze, die High-Level Descriptors abfragen, sind ebenfalls angedacht, genauso wie die Suche nach Dokumenten, die einem bestimmten Beispiel entsprechen (vgl. BLANKEN, BLOK ET AL. 2007, S. 13ff).

5.4.2 Browsing⁴⁰

Das „Browsing“ impliziert, dass der Nutzer keine genaue Vorstellung von dem hat, wie das Ergebnis seiner Suche konkret aussehen könnte. Er kann für sich jedoch eine Relevanz erkennen, wenn er ein Angebot an Dokumenten erhält. Generell besteht das „Browsing“ aus einer sehr allgemeinen Anfrage, die einem Stöbern in der Ergebnisliste vorangeht (vgl. BLANKEN, BLOK ET AL. 2007, S. 16). Dies ist besonders bei der Suche von Bild- und Videodokumenten sehr beliebt, da Webseiten, wie z.B. YouTube oder MyVideo kleine repräsentative Icons und eine kleine Inhaltsangabe für die gefundenen Videos anbieten. Der Nutzer kann dann selbst entscheiden, welcher seiner ermittelten Treffer für ihn wichtig sein könnte.

⁴⁰ Engl. Stöbern

5.5 Ergebnisqualität: Precision und Recall

Die Ergebnisqualität einer Suche ist für den Nutzer ein u.U. entscheidendes Kriterium, einen Suchdienst auch in Zukunft zu benutzen. Relevante Dokumente sollen von nicht-relevanten Dokumenten (Ballast) durch seine Anfrage getrennt werden.

Die Dokumente, die als relevant erkannt und auch tatsächlich gefunden wurden, werden als „Precision“ errechnet. Beim Recall werden die relevanten Dokumente zugeordnet, die auch tatsächlich gefunden wurden (vgl. BAEZA-YATES, RIBEIRO-NETO 2006, S. 75ff).

$$Pr ecision = \frac{r}{n}$$

$$Re call = \frac{r}{R}$$

r: Anzahl der gefundenen, relevanten Dokumente
n: Anzahl der gefundenen Dokumente
R: Gesamte Anzahl der relevanten Dokumente

6. Konzeption eines auf OWL-basierenden Retrievalsystems

Nachdem nun in den ersten fünf Kapiteln dieser Bachelorarbeit eine Einführung in mehrere relevante Standards, Methoden und Modelle stattgefunden hat, soll nun dieses Kapitel die Prinzipien eines Konzepts für ein Retrievalsystem für multimediale Metadaten auf Basis von OWL dargestellt werden. Neben einem Ontologieentwurf für dieses System soll außerdem auf weitere mögliche Teile eingegangen werden. Das wären z.B. eine passende Methode der Inhaltsextraktion und Gewinnung der Metadaten, eine Entscheidung für ein oder mehrere Retrievalmodelle und natürlich eine Evaluation der Realisierung eines solchen Systems.

Wie in den vorherigen Kapiteln bereits mehrfach angesprochen, ist die Bandbreite an Standards, Methoden und Modellen sehr vielfältig. Daraus kann man schließen, dass nicht jeder Standard auch der passende für jeden Anwender ist. Aufgrund dessen sollen zwei Varianten des Retrievalsystems präsentiert werden. Die erste Variante soll hauptsächlich für die Verwaltung multimedialer Metadaten von Privatanwendern gedacht sein. Die zweite Variante hingegen soll einem höheren Umfang enthalten und durchaus für komplexere multimediale Sammlungen zur Anwendung kommen. Die grobe Architektur dieses Retrievalsystems soll für beide Varianten gültig sein. Unterschiede ergeben sich lediglich innerhalb der einzelnen Bausteine des folgenden Modells. In den weiteren Unterkapiteln werden die Unterschiede genauer erläutert.

Anwenderszenarien sollen die Interessen konkreter darstellen und die Anforderungen an die verschiedenen Varianten formulieren.

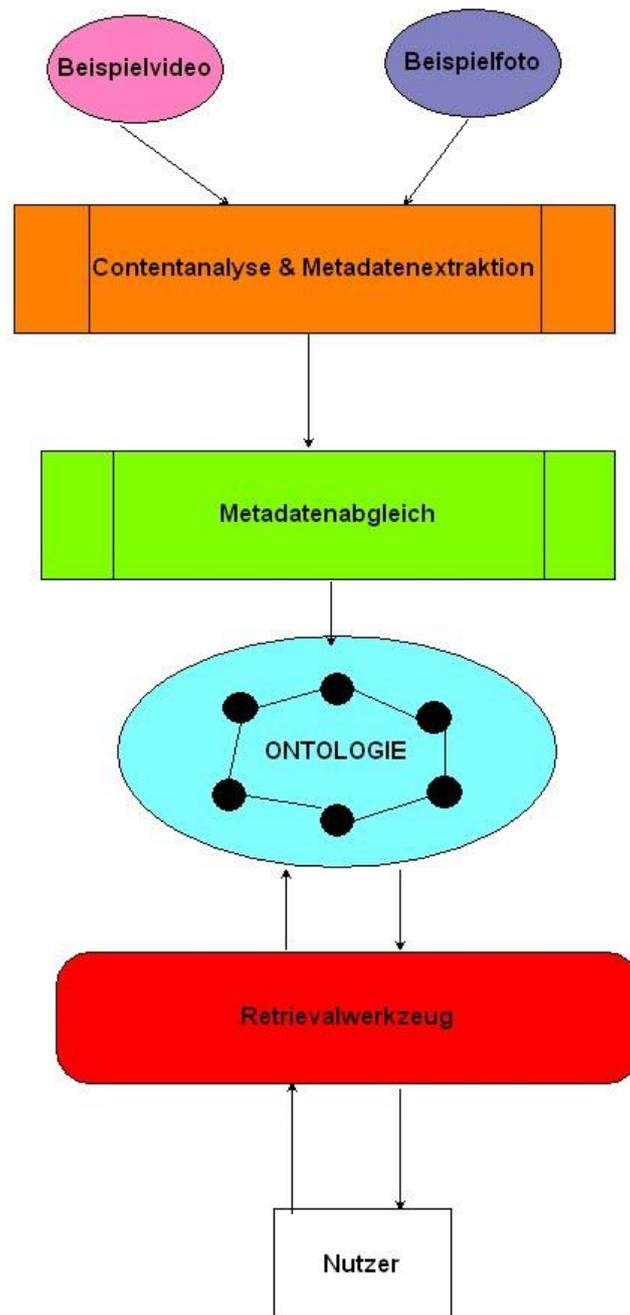


Abb. 12 Allgemeine Architektur des Retrievalsystems

6.1 Ontologie Retrievalsystem für Privatanwender

6.1.1 Anwenderszenario

Der Nutzer eines solchen Retrievalsystems möchte seine privaten Fotos und Videos verwalten. Dazu können u.a. Fotos und Videos aus dem Urlaub, von den Kindern etc. gehören. Da sich mittlerweile auch gerade für Privatanwender die Erstellung von multimedialem Content deutlich vereinfacht hat, möchte er sich ein kleines Archiv aufbauen, in welchem er die eigenen Erzeugnisse schnell und unkompliziert wieder findet. Anders formuliert, es soll ein Mittel zum Zweck sein.

Da der Inhalt des Systems hauptsächlich privat und daher nur von einem kleineren Kreis von Nutzern angewendet wird, spielt eine zu genaue Objektivität der Metadaten keine große Rolle. Auch soll möglichst ein einfacher Zugang ermöglicht werden, unabhängig von Kenntnissen in Abfrage- bzw. Retrievalsprachen. Jedoch möchte er von einer ontologiebasierten Suche profitieren und nicht mehr auf eine formale Suche angewiesen sein.

6.1.2 Mögliche technische Realisierung

Zu Beginn der Verwendung dieser Privatanwendervariante steht die Verwaltung des multimedialen Content. Während die multimedialen Dokumente an sich auf den jeweiligen Datenträgern gespeichert und nachgewiesen sind, ist für die Verwaltung der Metadaten eine Ontologiedatenbank notwendig.

Die Beschreibung und inhaltliche Erfassung eines Dokuments kann hier zunächst durch die Annotation eigener Metadaten erfolgen. Die formalen Daten könnten dann automatisch erschlossen werden. Ein ähnliches Konzept bietet die Software „Caliph“, die auf MPEG7-Basis entwickelt wurde (vgl. LUX 2007). Hier kann der Nutzer ohne großen Aufwand eine

Beschreibung für ein Fotodokument erstellen und zusätzlich dazu Verbindungen zwischen den dazugehörigen Schlagwörtern, auf Grundlage eines MPEG7-Descriptor Schemes, schaffen. Nachteil dieses Programms ist die nichtvorhandene Unterstützung von Ontologien, die eine automatische Verknüpfung zwischen Schlagwörtern ermöglichen könnten und die ausschließliche Bildannotation. Das hier konzeptionierte Retrievalsystem soll neben Bildern auch das Videoretrieval ermöglichen.

Zur Erfassung einiger formaler Daten könnte die Anwendung eines Teils des MPEG7-Standards durchaus hilfreich sein. Bereits existierende Low-Descriptors des Standards, wie z.B. Angaben zur Größe, Auflösung oder auch Farben etc. des Dokuments könnten angewendet werden (vgl. MANJUNATH 2002, S. 195 ff).

Neben einer konventionellen Suche über Abfragen, sollte neben den gefundenen Ergebnissen auch ein Angebot von „ähnlichen“ Dokumenten präsentiert werden. Dabei soll es um Dokumente handeln, deren Metadaten in einer Beziehung innerhalb der verwendeten Ontologie stehen. Die Modellierung der Ontologie könnte OWL Lite zur Grundlage haben. Diese Teilsprache besitzt einen Umfang, der sich perfekt für Privatanwender eignet (s. Kapitel 3). Alternative wäre OWL DL, mit einem umfangreicheren Umfang an Inferenzmechanismen und keiner Reduzierung der Kardinalitäten auf binäre Entscheidungen. OWL Full Ontologien stehen definitiv nicht zur Disposition, da diese bislang kaum in Anwendungen eine Rolle spielten und so komplex sind, dass es durchaus keine Entscheidung im Suchprozess geben kann.

6.2 Ontologie Retrievalsystem für professionelle Anwender

6.2.1 Anwenderszenario

Im Gegensatz zum Anwenderszenario für ein privates Retrievalsystem, soll hier die Professionalität im Vordergrund stehen. Ein solches System wäre z.B. ein virtuelles Bildarchiv, welches diversen Journalisten Zugang zu verschiedenstem Bild- und Videomaterial für Printprodukte und Webcontent bietet. Eine Sammlung dieser Art müsste daher eine möglichst objektive und unabhängige Contentbeschreibung enthalten. Der Zugang soll möglichst einer großen Interessengruppe ermöglicht werden, die durchaus das Interesse haben, möglichst schnell die gesuchten Dokumente zusammenzustellen. Eine einfache Annotation wäre dementsprechend nicht angemessen und deutlich zu aufwendig. Eine Erfassung von neuen Dokumenten sollte hier daher eher (semi-)automatisch erfolgen. Bei Bedarf wäre eine Überprüfung durch eine Fachkraft sinnvoll, um einen korrekten Ablauf der Erfassung zu garantieren. Der Einsatz einer ontologieunterstützten Metadatenverwaltung wäre hier ein großer Vorteil, da ein optimiertes Retrieval auch bei unverhältnismäßig großen Dokumentmengen möglich wäre. Ein solches System wäre im Übrigen nicht auf einen einzelnen Rechner reduziert, sondern durchaus intranet- und webtauglich.

6.2.2 Mögliche technische Realisierung

Wichtigster Unterschied zum System für Privatanwender wäre die standardmäßige Verwendung von OWL DL als Modellierungssprache der Ontologien. OWL DL bietet einen größeren Modellierungsumfang und ermöglicht daher auch einen komplexeren Umgang mit einem Retrievalsystem dieser Art. Am Anfang des Prozesses steht auch hier die Erfassung des Inhalts des angebotenen Dokuments. Hier wäre ein Schlussfolgerungsmechanismus hilfreich, der aus den verschiedenen Low-

Level Informationen, wie z.B. Texturen, Farben, Formen etc. auch High-Level Informationen zusammensetzt und daraus die passenden Metadaten extrahiert. Anschließend kann eine Klassifizierung der Metadaten erfolgen und die Einbindung in die Ontologie kann erfolgen (CAVAZZA, GREEN ET AL. 1998, S 41). Eine andere Möglichkeit wäre die Verwendung von „Machine Learning“. Hierbei handelt es sich um ein automatisches Erkennungssystem, dass anhand von mehreren Erkennungsprozessen die semantischen Verbindungen zwischen den Metadaten erkennt, speichert und für weitere Erkennungsprozesse wieder anwendet. Technische Grundlage eines solchen Systems könnte z.B. ein Bayesian Netzwerk sein, dass kontinuierlich erweitert wird (vgl. STAMOU, KOLLIAS 2005, S. 237ff). Diese Klassifizierung wäre mit dem Metadatenabgleich in der Abb. 12 gleichbedeutend. Der Nutzer sollte auch in diesem Fall die Möglichkeiten des „Searching“ und „Browsing“ haben. Das „Searching“ könnte hier allerdings um die Möglichkeit von Anfragen in natürlicher Sprache erweitert werden. Die Verwendung des MPEG-7 Standards sollte auch hier realisiert werden. Der Einsatz von Ontologien würde den Standard auch deutlich zugänglicher und flexibler gestalten (ATHANASIADI, TZOUVARAS ET AL. 2005, S.59).

6.3 Ontologiemodellierung

6.3.1 Allgemeines

Die Modellierung von Ontologien für die Retrievalsysteme, die in den Unterkapiteln zuvor vorgeschlagen wurden, muss differenziert realisiert werden. Neben spezifischen Domainontologien, die Wissen eines bestimmten Themengebietes enthalten, ist eine übergeordnete Multimediaontologie von Vorteil. Eine solche übergeordnete Ontologie ermöglicht zum einen die Interoperabilität zwischen verschiedenen Systemen in semantischer und syntaktischer Hinsicht und zum anderen

ermöglicht diese Konstruktion ein geordnetes Verwalten der Gesamtheit aller verwendeten Ontologien (ARNDT, STAAB ET AL. 2007, S.5f).

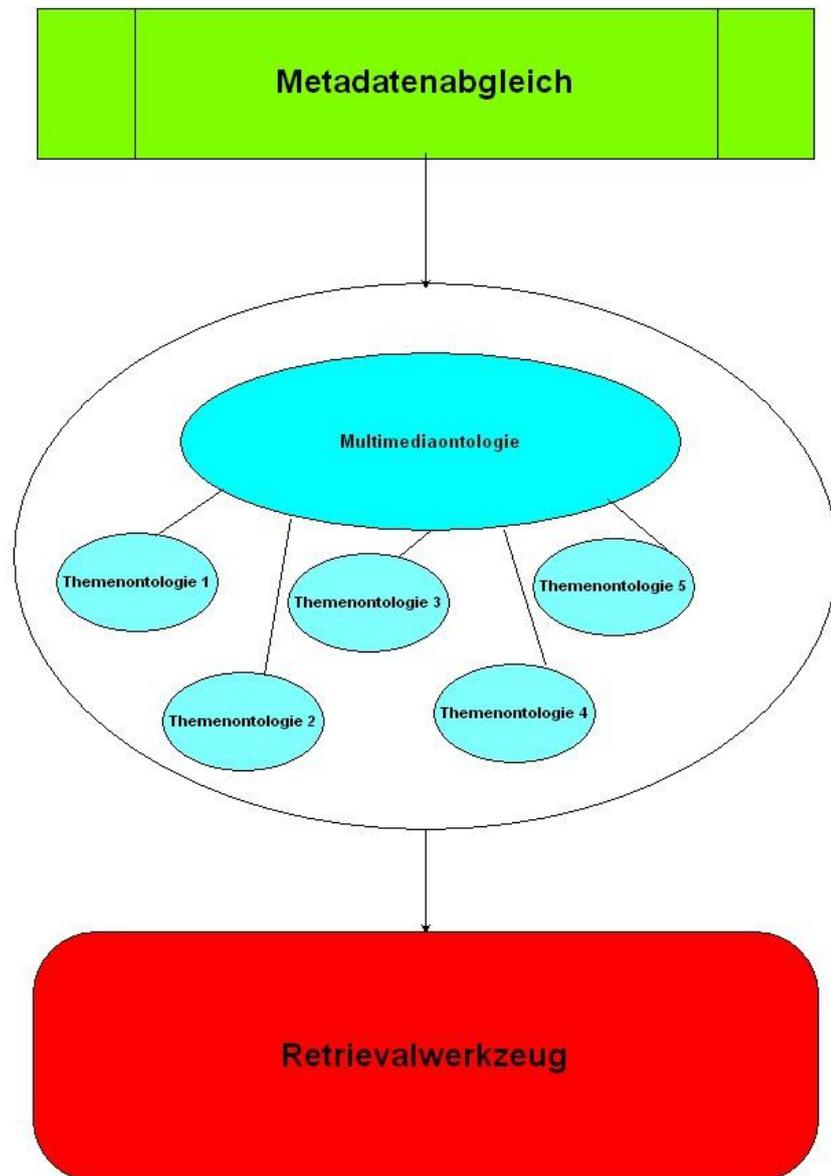


Abb. 13 Allgemeines Konzept der Ontologiearchitektur

Die Modellierung einer Ontologie und deren Klassen können nach folgenden Prinzipien erfolgen: 1. top-down Modellierung, 2. bottom-up Modellierung, 3. middle-out Modellierung (vgl. HÜSEMANN 2005, S. 71). Die top-down Modellierung berücksichtigt hauptsächlich das allgemeine Konzept der Ontologie, während sich die bottom-up Modellierung auf die Details eines Ontologiekonzepts fokussiert. Die middle-out Variante ist diejenige, die in der Praxis am häufigsten angewendet wird, da sie im Wechsel die beiden erstgenannten Methoden kombiniert. Jedoch muss betont werden, dass es keine allgemeingültige Methode zur Modellierung von Ontologien gibt. Es ist durchaus denkbar, dass es mehrere Wege gibt eine korrekte Ontologie zu erstellen. Bevor jedoch Klassen erstellt werden können, spielen die Abgrenzung des Ontologiegebietes und die dazugehörigen Begriffe eine wichtige Rolle. Das Themengebiet dieser Ontologie (Domain) sollte klären, welche Informationen sie abdecken soll, wofür sie Anwendung findet, welche konkreten Fragen sie beantworten kann und wie der Nutzerkreis aussieht. Eine weitere Hilfe zur Eingrenzung bieten auch sog. Kompetenzfragen, die mit Hilfe der Ontologie beantwortet werden sollen (NOY, MCGUINNESS 2001, S. 5). Kompetenzfragen zur übergeordneten Multimediaontologie in diesem konkreten Fall wären z.B. „Welche Fotos haben den Urheber Peter Schindler?“ oder „Welche Fotos können welchen Videos thematisch zugeordnet werden?“

Es ist ebenfalls hilfreich eine Liste von womöglich relevanten Begriffen für die Ontologie zu notieren. Diese sollen zum einen, einen Überblick über kommende Klassenstrukturen geben aber auch u.U. Überschneidungen bei der Darstellung von Konzepten verhindern und zum anderen kann so ein möglicher Erklärungsbedarf bei dem einen oder anderen Begriff aufgezeigt werden (NOY, MCGUINNESS 2001, S. 6).

6.3.2 Umsetzung Anhand einer einfachen Beispielontologie

Den Kern der Multimediaontologie dieses Konzeptes bildet zunächst die Superklasse „Metadaten“.

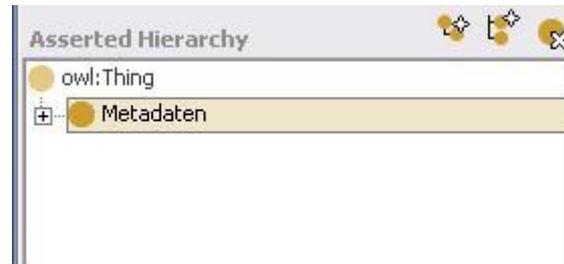


Abb. 14 Superklasse „Metadaten“

Es handelt sich dabei um eine abstrakte Klasse, da sie direkte Instanzen enthalten kann. Es existiert auch die Möglichkeit konkrete Klassen zu bilden, die unmittelbare Instanzen enthalten. (vgl. SACHS 2006, S. 21f.). Anders als in Dateiverwaltungssystemen werden in der Ontologie ausschließlich die Metadaten gespeichert. Die Klasse „Thing“ ist wie bereits in Kapitel 3 angesprochen, eine Standardklasse in OWL, die die Superklasse aller OWL Klassen bildet.

Die erste Ebene unterhalb der Superklasse „Metadaten“ bilden die Klassen „Formale_Metadaten_Foto“ und „Formale_Metadaten_Video“.



Abb. 15 Erste Subklassen

Die Klassen „Formale_Metadaten“ tragen die Namenszusätze des jeweiligen Dokuments, dessen Informationen zugeordnet werden sollen. Dies hat den Grund, dass sich die enthaltenen Instanzen vor allem im

formalen Bereich unterscheiden. Ein Fotodokument hat z.B. eine andere Dateiendung als ein Videodokument. Es handelt sich auch hier um abstrakte Klassen. Ergänzt werden die Informationen mit den abstrakten Subklassen „Visuelle_Metadaten_Foto“ und „Visuelle_Metadaten_Video“, welche wiederum die abstrakten Subklassen „Thematische_Metadaten_Foto“ und „Thematische_Metadaten_Video“ enthalten.



Abb. 16 Weitere abstrakte Subklassen

Am Ende dieser Klassenhierarchien stehen jeweils die konkreten Klassen „Foto“ und „Video“. Diese sind hier zunächst die alleinigen Instanzenträger.

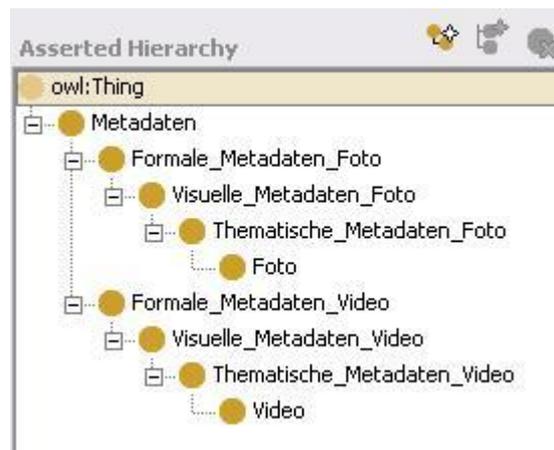


Abb. 17 Konkrete Klassen

Damit die Instanzen Informationen enthalten können, werden für die verschiedenen Klassen sog. Slots angelegt. Diese Slots haben die Aufgabe bestimmte Werte zuweisen zu können.

Die Klasse „Formale_Metadaten_Foto“ enthält daher wichtigen Slots, die eine Identifizierung über die Formalen Daten eines Dokuments möglich machen. Das gleiche gilt, für die Klassen „Visuelle_Metadaten“ und „Thematische_Metadaten“ mit jeweils eigenen Slots.



Abb. 18 Auflistung aller Slots

Zum besseren Verständnis und Übersicht der Klassen und ihrer jeweiligen Slots soll nun in tabellarischer Form dargestellt werden, wie die konkreten Verbindungen im Detail aussehen.

Klasse	Slots
Metadaten	-keine-
Formale_Metadaten_Foto & Video	Album (string), Dateiname(string), Dateiumfang (float), Dokumenttitel (string), Erstellungsdatum (string), Kameramodell (string)
Visuelle_Metadaten_Foto	Mpeg7_Color (integer), Mpeg7_Texture (integer)
Visuelle_Metadaten_Video	Mpeg7_Motion (integer), Mpeg7_Color (integer)
Thematische_Metadaten_Foto & Video	Systematik (string), Schlagworte (string)
Foto & Video	-alle Slots der übergeordneten Klassen-

Tab. 4 Klassen und Slots

Neben einer Einstellung des Wertebereichs, ist auch die Definition von Kardinalitäten möglich. Während Slots, wie z.B. Dateiname, Dokumenttitel oder Erstellungsdatum einmalig pro Instanz auftauchen sollen, können Schlagwörter und Systematiken mehrfach vergeben werden. Der Slot Kameramodell ist hingegen optional, wenn er jedoch verwendet wird, dann genau einmal.

„Mpeg7_Motion“ sollen verdeutlichen, dass die Ontologie über eine Schnittstelle MPEG-7 Descriptors erkennen und zu einer besseren Objektivität beitragen soll. Selbstverständlich lassen sich die Slots erweitern, sodass weitere MPEG-7 Elemente eingebaut werden können. Die folgende Abbildung zeigt die Erstellung einer Beispielinstantz.

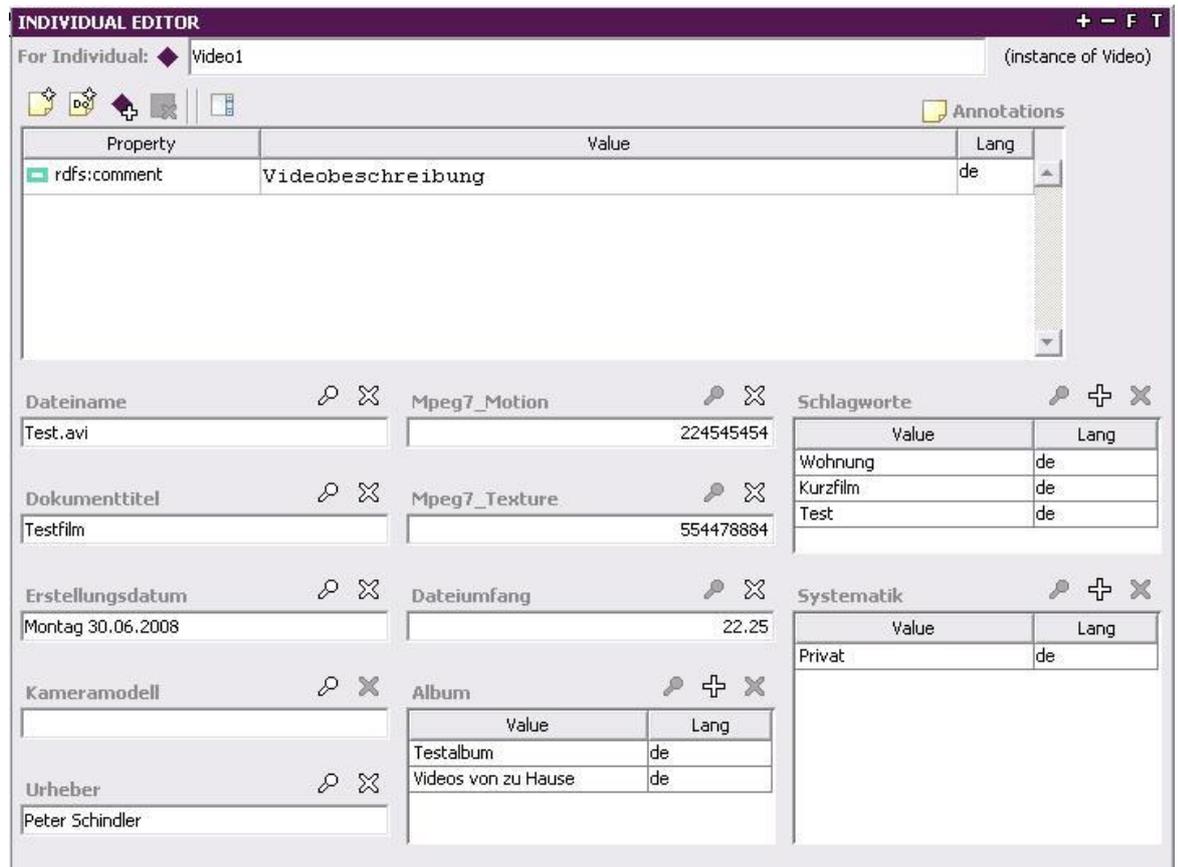


Abb. 19 Beispielinstanz

Das Beispiel zeigt, dass sich dieser Aufbau der Ontologie zu einer Annotation von Foto- und Videodateien eignet. Die Werte der angedeuteten MPEG-7 Descriptors sind hier nur willkürliche Zahlen.

6.3.3 Evaluation der Ontologie

Die Gestaltung der Ontologie richtete sich in diesem Beispiel nach der „Top-Down-Methode“. Durch die unterschiedlichen Felder ist es möglich, dass der Nutzer verschiedene Bilder erfassen und diese dann in einem Archiv verwalten kann. Durch die bestimmten Bedingungen, die den Slots zugewiesen wurden, ist der Nutzer zwar an bestimmte Vorgaben gebunden, was ihm jedoch dient, da falsche Eingaben so unterbunden werden können, daher Fehler minimiert werden können. Zusätzlich könnten noch Slots für

Beziehungen zwischen einzelnen Instanzen, in diesem Fall Beziehungen zwischen Fotos und Videos, integriert werden, um einen stärkeren semantischen Gehalt zu erzielen. Hier sollte jedoch der Nutzer individuell die Möglichkeit bekommen, diese selbst zu produzieren und zu integrieren. Des Weiteren muss noch eine Schnittstelle zu einer Applikation geschaffen werden, die die passenden MPEG-7 Descriptor Werte extrahiert und in die Ontologie integriert. Die Tatsache, dass die Zuweisung von Systematiken und Schlagworten manuell durch den Nutzer geschieht, zeigt zwar, dass es durchaus zu subjektiven Angaben kommen kann, dies ist jedoch nicht unbedingt zu kritisieren. Im Nutzerszenario wurde bereits erwähnt, dass die Nutzung der Daten nur für einen überschaubaren Nutzerkreis, wie z.B. Familienmitgliedern angedacht ist. Generell zeigt diese Ontologie ebenfalls, dass es durchaus simpel sein kann solche (einfachen) Beziehungen zu erstellen. Neben reinen MPEG-7 Anwendungen, die hauptsächlich auf XML-Schema beruhen und daher semantische Beziehungen nur in einer begrenzten Ausprägung möglich sind, hat OWL hier ein deutlich höheres Potential.

6.3.4 Umsetzung Anhand einer komplexeren Themenontologie

Ein solches Annotationswerkzeug auf OWL-Basis hat jedoch bestimmte Grenzen, wie z.B. eine zu große Subjektivität bei der Verschlagwortung des multimedialen Contents. Folgend soll nun eine Möglichkeit präsentiert werden, wie man dieses Problem lösen könnte. Aufbauend auf der Abb. 13 sollen deshalb Themenontologien (engl. Domain Ontologies) zum Einsatz kommen. In diesem Falle sollten es Ontologien sein, die geeignetes Wissen eines Pressearchivs abdecken. Dies bedeutet, dass die Beschreibung des multimedialen Contents nicht aus zwei Slots besteht, wie im vorherigen Beispiel verdeutlicht, sondern die Beschreibung besteht aus ganzen Klassen- und Beziehungsgebilden. Eine Teil der formalen Informationen der multimedialen Dokumente wird ebenfalls in eingebettet. Diese sind dann Teile der einzelnen Instanzen.

Die wichtigsten Superklassen der Ontologie sollen die verschiedenen Sparten z.B. einer Zeitung verdeutlichen. Das Pressearchiv soll zur besseren Orientierung ebenfalls aus diesen Sparten bestehen.



Abb. 20 Superklassen Themenontologie

Diese Superklassen sind abstrakte Klassen. Im Folgenden soll am Beispiel einer einzelnen Superklasse (Superklasse „Politik“) erläutert werden, wie das Wissen und die Metadaten im Detail organisiert sind. Dieses Funktionsprinzip funktioniert für alle Superklassen dieser Ontologie. Die Superklasse „Politik“ ist drei abstrakte Subklassen der ersten Ebene unterteilt.



Abb. 21 Subklassen der ersten Ebene

Die Subklasse „Deutschland“ ist hier besonders detailliert modelliert. Dies hat den praktischen Grund, dass das Wissen über die Inlandspolitik, also hier die deutsche Politik, im Gegensatz zum Wissen der ausländischen Politik meist eine größere Detailtiefe haben muss. In einem konkreten

Beispiel erklärt, bedeutet das, dass die Informationen und Metadaten zur Deutschen Bundesregierung dieser Ontologie ausführlicher sind als die zur Administration der Vereinigten Staaten, da es de facto deutlich wahrscheinlicher ist, dass in einer deutschen Tageszeitung ein Foto eines deutschen Staatssekretärs gebraucht wird, als das eines US-amerikanischen.

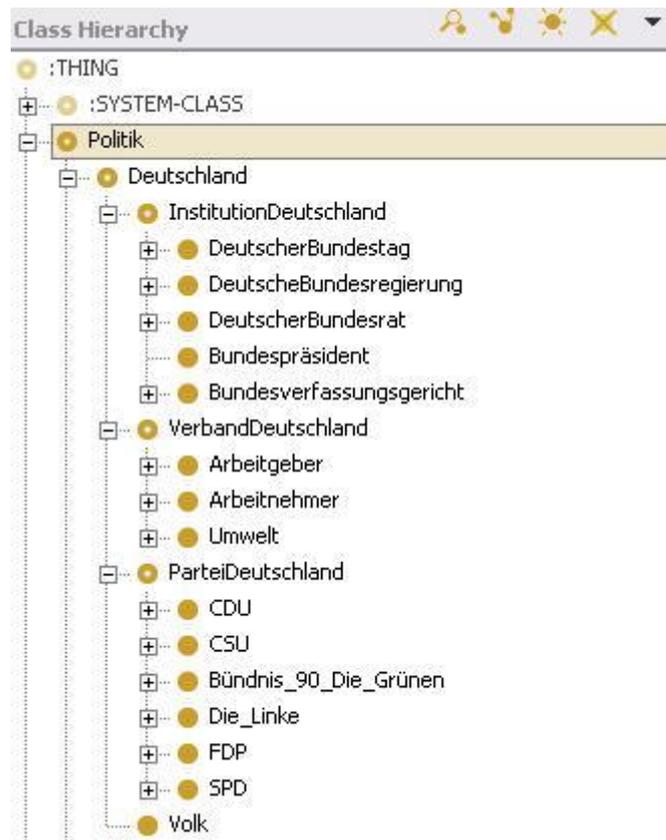


Abb. 22 Weitere Subklassen der Superklasse Politik und der ersten Subklasse Deutschland

Zur Verdeutlichung des Gebrauchs von Instanzen sollen nun die Subklassen „DeutscheBundesregierung“ und „CDU“ im Fokus stehen.

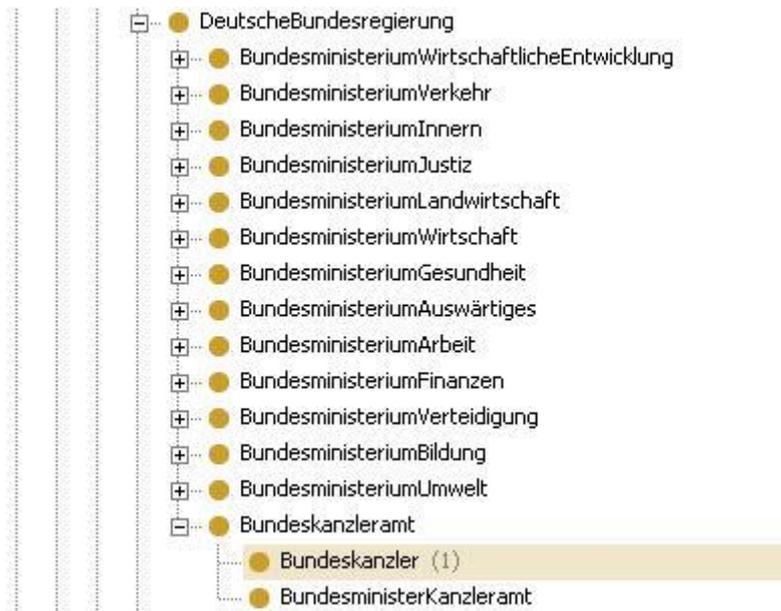


Abb. 23 Instanzbeispiel Teil 1

Die Klasse „DeutscheBundesregierung“ unterteilt sich in die einzelnen Klassen, die die Ministerien repräsentieren. Diese wiederum haben eine Klasse für einen Minister und einen Staatssekretär. Im Fall der Klasse „Bundeskanzleramt“ ist es die Klasse „Bundeskanzler“ und „BundesministerKanzleramt“. Die Klasse „Bundeskanzler“ enthält bereits eine Instanz.

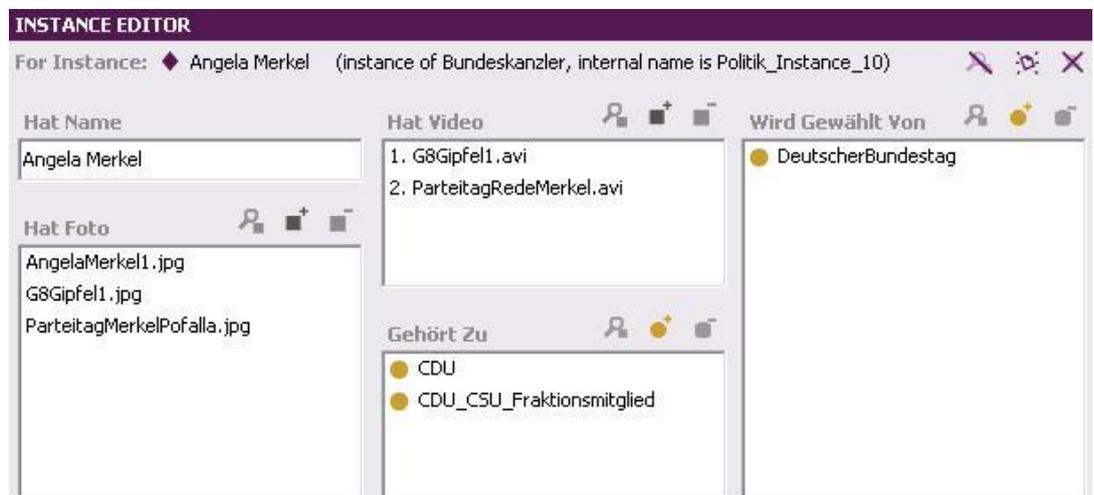


Abb. 24 Instanz der Klasse „Bundeskanzler“

Die Instanz der Klasse „Bundeskanzler“ hat den Namen „Angela Merkel“. Darüberhinaus beinhaltet diese Instanz Informationen zu den Verbindungen zu anderen Klassen. „Angela Merkel“ ist nicht nur Teil der Klasse „Bundeskanzler“ sondern auch Teil der Klassen „CDU“ und „CDU_CSU_Fraktionsmitglied“. Des Weiteren existiert die Information, dass „Angela Merkel“ von der Klasse „DeutscherBundestag“ gewählt wurde. Die Slots „Hat_Foto“ und „Hat_Video“ stellen hier Strings dar. Diese beiden Slots sind für alle konkreten Klassen dieser Ontologie vorhanden. Es soll daran erinnert werden, dass diese Ontologie zwar durchaus eine umfangreiche thematische Wissensstruktur enthält, jedoch in erster Linie zur Verwaltung von multimedialen Dokumenten dienen soll. Protégé bietet die Möglichkeit an bereits vorhandene Projekte einzubinden, so dass die Klasse „Metadaten“ aus Kapitel 6.3.2 in modifizierter Form durchaus integriert werden könnte und die einzelnen Instanzen dieser Klasse auch z.T. Relationen zu Instanzen der Klasse „Bundeskanzler“ haben könnte.

Die Instanz „Angela Merkel“ ist jedoch nicht nur in der Klasse Bundeskanzler zu finden, sondern auch in der Klasse „CDU“. Es soll verdeutlicht werden, dass „Angela Merkel“ neben dem Amt des Bundeskanzlers auch das Amt einer Parteivorsitzenden ausübt.



Abb. 25 Instanzbeispiel Teil 2

Dementsprechend gehört die Instanz „Angel Merkel“ zur Klasse „ParteyvorsitzCDU“. Um noch zu verdeutlichen, dass der Parteivorsitz durch die Klasse „ParteitagCDU“ gewählt wird, wird zum Slot

„Wird_Gewählt_Von“ der Instanz „Angela Merkel“ noch die Klasse „ParteitagCDU“ hinzugefügt.

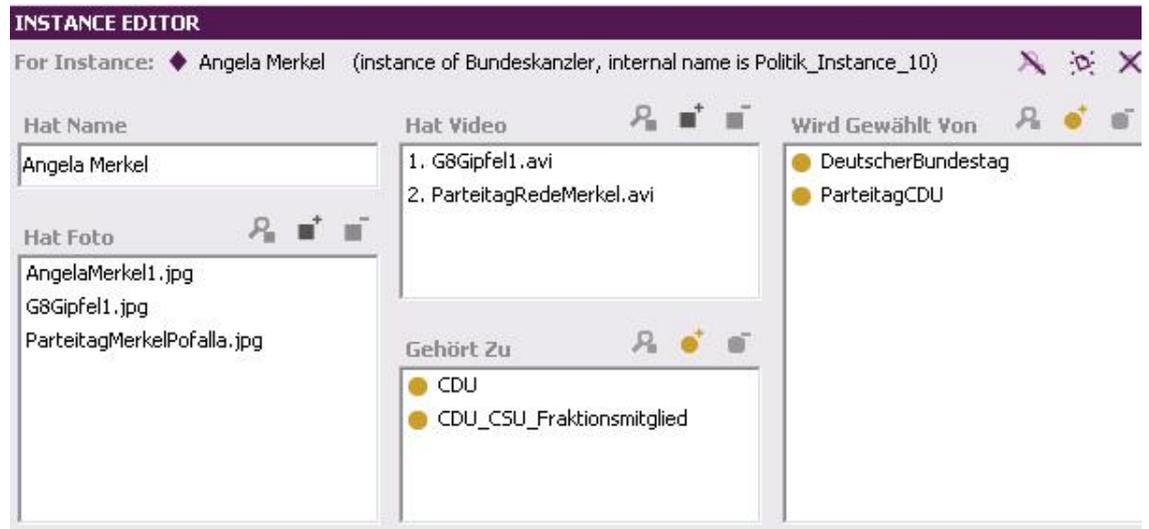


Abb. 26 Instanz der Klasse „Bundeskanzler“ und „ParteivorsitzCDU“

6.3.5 Evaluation der Ontologie

Aufgrund der deutlichen Komplexität dieser Ontologie gestaltete sich die Modellierung deutlich schwieriger. Eine middle-out-Variante zur Modellierung erwies als am besten geeignet, da die Klassentiefe im Gegensatz zur einfachen Ontologie deutlich detailreicher war. Deutliche Vorteile durch die Verwendung von OWL können in so einer Themenontologie zur Geltung kommen. Inferenzmechanismen können bestimmte Schlüsse ziehen, was über eine reine hierarchische Ordnung, wie z.B. in einer Verzeichnisstruktur eines Dateisystems hinausgeht. Das Vermeiden von subjektiver Verschlagwortung kann durch eine festgelegte Struktur umgangen werden und erleichtert daher eine Suche von Nutzern, die nicht auch Entwickler dieser Architektur sind. Darüber hinaus ist durch die Kombination der Metadatenontologie mit den Themenontologien neben einer vielschichtigeren thematischen Systematisierung auch eine offensichtlich umfangreichere formale Beschreibung der einzelnen Dokumente realisierbar.

Der große Umfang des thematischen Gebietes gestattet ferner den Einsatz sämtlicher Möglichkeiten, die OWL bietet. Ein Beispiel wäre die inverse Nutzung von Slots. Neben dem Slot „Wird_Gewählt_Von“, existiert parallel auch der Slot „Wählt“.

6.4 Funktionskonzept Inhaltsextraktion

Wie bereits vorher genannt, beinhaltet die komplexe Ontologie keine manuelle Annotation der multimedialen Dokumente. Um die Objektivität zu steigern soll eine (semi-)automatische Extraktion der Metadaten erfolgen. In mehreren Schritten soll erläutert werden, wie der Ablauf von einem einfachen Foto zum fertigen Retrievalprodukt ablaufen könnte. Am Beispiel eines Fotos der Bundeskanzlerin Merkel soll dieses Beispiel demonstriert werden.



Abb. 27 Ausgangsbild von Bundeskanzlerin Angela Merkel (vgl. HEIN 2008)

Zu Beginn dieses Funktionsablaufs steht die Contentanalyse. Hier würde sich die Anwendung von MPEG-7 Descriptors anbieten. Zunächst werden formale Daten, wie z.B. die Auflösung, Farb- und Texturdaten, aber auch ein möglicher Titel und ein Erstellungsdatum erfasst. Diese Werte können direkt extrahiert und zwischengespeichert werden, da zunächst noch die inhaltliche Analyse des Bildes folgen muss. Der erste Schritt hierbei wäre eine Kantenabgrenzung (Konturanalyse) um das im Bild enthaltene und zu erkennende Objekt zu identifizieren. Darüber hinaus erfolgt noch eine Aufteilung in eine Farb- und eine Texturanalyse.



Abb. 28 Erste Analyse des Ausgangsbildes

Aus den ersten low-level Werten könnte geschlossen werden, dass es sich um eine Person handelt. Hier könnte nun die Abrufung einer Personendatenbank erfolgen, die die enthaltene Person durch ein Gesichtserkennungsprogramm, wie sie z.B. die Eigenobjects-Erkennung (s. Kapitel 4.2.1) bietet, erkennt. Ist dieser Vorgang vollzogen, kann eine

festgelegte Anzahl an Schlagwörtern erstellt werden, die u.U. ebenfalls in einer Datenbankabfrage erstellt und in Verbindung zur erkannten Person gebracht wurden. Der nächste Schritt wäre der Abgleich der Schlagwörter mit der Ontologie. Ist eine identische Klasse gefunden, können die zwischengespeicherten formalen und die nun dazugehörigen inhaltlichen Informationen in eine thematische Klasse und die Fotoklasse integriert werden.

Eine Schwierigkeit wäre jedoch die Unterscheidung in relevante und nicht-relevante Objekte, die erkannt werden. Im Hintergrund ist ein Instrument zu sehen, welches keine Rolle spielen soll. Eine Möglichkeit wäre, dass am Ende des Erkennungsvorgangs nicht-relevante Objekte manuell ignoriert werden. Dies hätte jedoch den Nachteil, dass bei einer großen Menge von Fotos auch ein dementsprechend großer Arbeitsaufwand die Folge wäre. Eine automatische Option wäre die Eingrenzung der Erkennung auf bestimmte Objekte. Nicht-relevante Objekte würden von vornherein aussortiert werden. Neben der Erkennung von Personen, die zumindest für solch eine Ontologie am relevantesten ist, spielt das Erkennen der entsprechenden Ereignisse und der dazugehörigen Orte ebenfalls eine wichtige Rolle. Ein Lösungsansatz wäre, dass Fotos einer bestimmten Serie von vornherein formal bezeichnet werden, so dass der Anlass der Fotos vor der Erstellung der Fotodokumente eindeutig zu ermitteln ist.

Der Ablauf bei Videodateien unterscheidet sich im Erkennungsprozess des Dokuments. Es erfolgt eine Einteilung in Szenen, die durch Schnitte getrennt werden (Shots). Bei harten Schnitten, wo ein eindeutiger Übergang zu erkennen ist, ist dies relativ leicht zu realisieren. Aufwendigere Techniken sind bei sog. weichen Schnitten nötig, wie z.B. Überblendungen (s. Kapitel 4.2.2). So gewonnene Low-Level Descriptors müssten nun ebenfalls mit Hilfe eines Datenbankabgleichs zu High-Level Descriptors „zusammengesetzt“ werden, um eine effektive Metadatenauslese zu realisieren.

6.5 Funktionskonzept Retrieval

Nach dem die Ontologie mit den für die Bild- und Videodokumenten repräsentierten Instanzen gefüllt wurde, steht nun das Retrieval dieser im Vordergrund. Dieses Funktionskonzept besteht aus zwei Säulen. Neben einem Abfragesystem, in dem der Nutzer seine Suchanfrage stellen kann, soll das Retrievalsystem ebenfalls einer Baumstruktur ähnlich sein, wie die der Protégé-Gestaltungsstruktur, die der Erstellung der Ontologie diene. Eine Beispielsuche nach Fotos von Angela Merkel könnte z.B. folgendermaßen aussehen. Eine Anfrage könnte damit beginnen, dass die Suche auf eine bestimmte Klasse beschränkt wird. Das wäre in diesem Fall, die Klasse „Politik“. Anschließend könnte ein Slot ausgewählt werden, um deutlich zu machen, nach was genau man sucht. Hier wäre es der Slot „Hat_Foto“, um eine Suche nach Fotometadaten zu ermöglichen. Anschließend könnte eine Bedingung hinzugefügt werden, wie z.B. „contains“ (Suchanfrage enthält) oder „begins“ (Suchanfrage beginnt mit). Zum Abschluss folgt dann die Eingabe des gesuchten Strings.



Abb. 29 Suchabfrage für die Fotos von Angela Merkel

Bei der Eingabe des Suchstrings „merkel“ unter der Einbeziehung der Bedingung „contains“ erhält der Nutzer nun sämtliche Instanzen, die den String enthalten. Darunter befinden sich sämtliche Fotos, die mit der Bundeskanzlerin zu tun haben. Aber da Angela Merkel darüberhinaus auch Parteivorsitzende und Mitglied der CDU/CSU-Fraktion ist, findet sich z.B. die Instanz „Angela Merkel“ dreimal in der Suchergebnisliste wieder. Ein Nebenprodukt der Suchabfrage nach den Fotos von Angela Merkel sind deshalb auch die zusätzlichen Informationen, die aus der Ontologie geschlossen wurden.

Das verwendete Retrievalmodell und das dazugehörige Werkzeug innerhalb des Protégé-Query Editors lassen sich mit einer SQL-Abfrage vergleichen. Boolesche Operatoren sind einsetzbar. Ebenso lässt sich die Suchanfrage an bestimmte Bedingungen knüpfen. Bei MySQL kann ein Volltext-Index auch aus bestimmten Tabellen bestehen und andere weitere Bedingungen erfüllen.

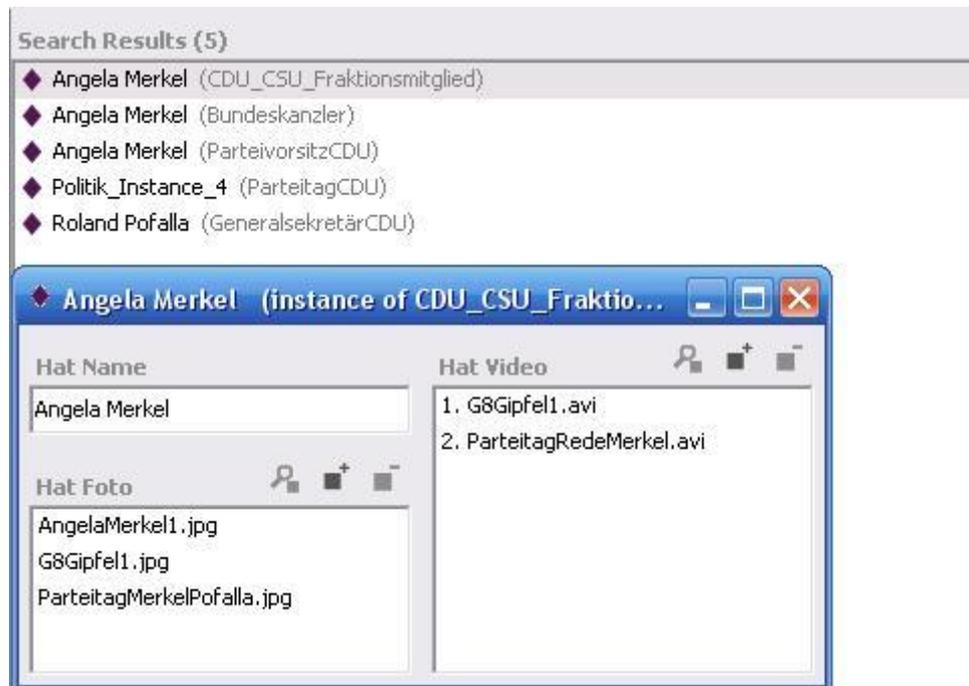


Abb. 30 Ergebnisliste und Inhalt der ersten Instanz

Die zweite Säule soll ein „Browsing“ ermöglichen. Die Baumstruktur der Ontologie soll für den Nutzer zugänglich sein. Dies hat den Vorteil, dass das repräsentierte Wissen transparent dargestellt ist. Ein Nutzer, der auf der Suche nach Bildern ist, jedoch keine Suchanfrage formulieren kann, hat daher die Option innerhalb der Ontologie nach passenden Instanzen zu stöbern. Diese erweiterte Suchmöglichkeit soll dafür sorgen, dass auch die Aussichten geeignete Ergebnisse zu finden, steigen. Um jedoch von vornherein sicher zu gehen, dass sich auch Nutzer mit geringen Ontologiekennnissen zurechtfinden, sollte die Benutzeroberfläche nicht der der Protégéoberfläche eins zu eins entsprechen. Die Bezeichnungen der

Klassen, Slots und der Instanzen sollten daher allgemeingültig sein, damit keine Missverständnisse bei der Suche auftreten. Hier könnte z.B. mit der Anzeige eines *rdfs:label* ein allgemeingültiger Name in der Nutzeroberfläche dargestellt werden (vgl. ALLEMANG, HENDLER 2008, S. 110).

Selbsterklärend ist die Möglichkeit für „einfache“ Nutzer die Ontologie in irgendeiner Art und Weise zu modifizieren nicht gegeben. Dies sollte einer angemessenen Zahl an Administratoren vorbehalten sein, die sich mit dem jeweiligen Themengebiet auskennen.

6.6 Fazit

Dieser Konzeptteil der Bachelorarbeit sollte eine Reihe von Vorschlägen sammeln, wie ein Retrievalsystem auf OWL-Basis funktionieren kann. Mit dem Ontologieeditor Protégé wurden zwei unterschiedliche Ontologien modelliert. Eine einfache zur Verwaltung von privatem Content und eine komplexere Themenontologie, die in modifizierter Kombination mit der einfachen Ontologie eine umfangreichere Sammlung beinhalten kann und diese besser zugänglich macht als gewöhnliche syntaktische Retrievalmodelle. Darüber hinaus wurden Komponenten zur Extraktion von Metadaten und Suchoptionen für ein potentielles Retrievalverfahren vorgestellt. Die sich an MySQL anlehende Möglichkeit von Suchabfragen innerhalb der Ontologie kann zu einem schnelleren und unkomplizierten Zugang führen, da SQL-Suchen auf vielen Webseiten seit längerer Zeit verwendet werden. Die Verwendung von anderen hilfreichen Standards, wie z.B. MPEG-7 sollte das Konzept um bereits existierende und sehr hilfreiche Tools ergänzen.

Generell muss angemerkt werden, dass der wichtigste Nutzen von OWL, neben einer relevanten Unabhängigkeit von bestimmter Software, die Kombination aus Content und Wissensrepräsentation ist. Die Grundlage für das Zusammenspiel der oberen Schichten aus dem Semantic Web

Schichtenmodell (Proof, Trust und Unifying Logic) ist mit einer OWL-Ontologie belegt und lässt Interpretationen für passende Tools zu. Eine Demonstration solcher Software ist jedoch nicht Thema dieser Bachelorarbeit und wäre eher im Fachgebiet Informatik angesiedelt, da die dementsprechende Kenntnis in Softwarearchitektur und fortgeschrittener Programmierung Voraussetzung ist. Anzumerken ist, dass sich die Anwendung von „Precision“ und „Recall“ bei ontologiegestützten Retrievalsystemen als unabhängiger Richtmesser für die Qualität des Retrievals als durchaus problematisch erweisen kann. Beide Indikatoren sind zwar in der Fachwelt allgemein akzeptiert, haben jedoch den Nachteil, dass Ergebnisse, die nah am gewünschten Resultat liegen, nicht gewürdigt werden und daher ebenso eine schlechte Bewertung bekommen können, wie z.B. Retrievalergebnisse, die den Anforderungen grundsätzlich überhaupt nicht entsprechen. Eine Evaluation ist daher mit einem erweiterten „Precision“ und „Recall“ möglich (vgl. EUZENAT, SHVAIKO 2007, S. 208ff). Darüber hinaus wäre eine komplette Beurteilung des Systems nur mit einer fertigen Applikation möglich, was jedoch wie bereits erwähnt, nicht Thema dieser Bachelorarbeit ist.

7. Zusammenfassung und Ausblick

Diese Bachelorarbeit sollte zunächst einmal einen Überblick über aktuelle Standards und Werkzeuge des Semantic Web geben. Das Semantic Web als Erweiterung und nicht als komplette Neuerfindung des World Wide Web stand dabei im Vordergrund und eine Verbindung zum Bereich des multimedialen Contents sollte gezogen werden. Dabei spielte vor allem die Vorstellung eines verbesserten Retrievalsystems für Video- und Fotodateien auf semantischer Basis im Fokus der Arbeit.

Die Ontologiesprache OWL sollte dabei der entscheidende Grundstein für solch ein Konzept sein, da sie im Gegensatz zu anderen Standardsprachen, wie z.B. XML oder RDF, einen weit größeren Umfang und daher auch einen komplexeren Gestaltungsspielraum ermöglicht. Darüber hinaus bot der Ontologieeditor Protégé eine benutzerfreundliche Gestaltungsoberfläche, mit der sich Ontologien ohne große Hindernisse modellieren ließen. Wie die Zukunft eines solchen semantischen Retrievalkonzepts und dessen tatsächliche Umsetzung konkret aussieht, wird sich wohl an dem tatsächlichen Bedarf der Nutzer orientieren. Zur Zeit beinhaltet das Web ca. 2 Prozent an semantischen Informationen (vgl. Spiegel ONLINE 2008), was beweist, dass diese Technologien definitiv noch Entwicklungs- und Erklärungsbedarf haben. Für eine stetige Entwicklung im multimedialen Bereich könnte die immer günstiger werdende Erstellung und Speicherung von Bildern und Fotos sorgen. Hier wäre ein Bedarf eines optimierten Retrievals erklärbar. Besonders im professionellen Bereich, wie z.B. Pressearchiven oder dem Medizinsektor wäre ein solches Konzept mit den dementsprechenden Ontologien eine weitreichende Hilfestellung. Gerade die Kombination aus „Searching“ und „Browsing“ wäre hier hervorzuheben, da nicht nur vage bzw. genaue Suchabfragen zu Ergebnissen führen würden, sondern auch Ansätze von systematischen Suchen in Wissensdatenbanken.

Erste Projekte dazu wurden bereits in der Einleitung erwähnt.

Schwierigkeiten könnten besonders im Bereich der korrekten Gestaltung

von Ontologien liegen. Bei komplexen Themenontologien werden Experten allgemeingültige Konzepte erarbeiten müssen, die damit einen Konsens repräsentieren, Objektivität garantieren und in Folge dessen die Zahl der Missverständnisse bei Suchverfahren minimiert wird. Hier könnte allerdings auch der wesentliche Vorteil der Interoperabilität zum Zuge kommen, da bereits vorhandene Ontologien miteinander verknüpft werden können. Die Standards sollen auch garantieren, dass die Anwendungen unabhängig von bestimmten Computerplattformen lauffähig sind. Der Einsatz zusätzlicher Standards, z.B. MPEG-7 sowie Tools zur automatischen Inhaltsextraktion erleichtern zudem das Archivieren der Dateien und das Erstellen der Metadaten.

Glossar

Agent	In diesem Fall ein Programm zur Suche von Webseiten, nach vom Nutzer bestimmten Kriterien.
Browsing	Stöbern
dashboard	Armaturenbrett
Desktop	Arbeitsoberfläche
DL	Description Logic, Beschreibungslogik
DTD	Document Type Definition
HTML	Hypertext Markup Language, Standardsprache zur Erstellung von Webseiten
integer	Nichtnegative Ganzzahl
Landmark	Hier: Abgrenzung der Form eines Objektes in einem Bild
MPEG	Moving Pictures Expert Group
Namespace	Namensraum
OWL	Web Ontology Language
Proof	Nachweis
RDF	Ressource Description Framework
RDFS	Ressource Description Framework Schema
Searching	Gezieltes Suchen
Semantik	Aus dem Griechischen bezeichnen, anzeigen
SQL	Structured Query Language, Abfragesprache für relationale Datenbanken
string	Zeichenkette

SWO	SemanticWebOntology
Syntax	Aus dem Griechischen die Zusammenstellung
topic map	Abstraktes Modell zur Visualisierung von Wissensstrukturen
Trust	Vertrauen
Unicode	ISO-Standard, enthält sämtliche Schriftzeichen aller Sprachen
Unifying Logic	Vereinigende Logik
Web of Trust	Netz des Vertrauens
XML	Extensible Markup Language

Literaturverzeichnis

Allemang, Hendler 2008

ALLEMANG, Dean ; HENDLER, James A.: *Semantic web for the working ontologist : Modeling in RDF, RDFS and OWL*. Amsterdam , Boston : Morgan Kaufmann Publishers/Elsevier, 2008. – ISBN 9780123735560

Arndt u. a. 2007

ARNDT, Richard ; STAAB, Steffen ; TRONCY, Raphaël ; HARDMAN, Lynda: *Adding Formal Semantics to MPEG-7: Designing a Well-Founded Multimedia Ontology for the Web*. Koblenz, 2007(Arbeitsberichte aus dem Fachbereich Informatik 4)

Athanasiadi u. a. 2005

ATHANASIADI, Thanos ; TZOUVARAS, Vassilis ; PETRIDIS, Kosmas ; PRECIOSO, Frederic ; AVRITHIS, Yannis ; KOMPATSIARIS, Yiannis: *Using a Multimedia Ontology Infrastructure for Semantic Annotation of Multimedia Content*. Athen, 2005

Baeza-Yates, Ribeiro-Neto 2006

BAEZA-YATES, Ricardo ; RIBEIRO-NETO, Berthier: *Modern information retrieval*. [Nachdr.]. Harlow : Pearson Addison-Wesley [u.a.], 2006. – ISBN 020139829x

Berners-Lee, Hendler, Lassila 2001

BERNERS-LEE, Tim; HENDLER, James; LASSILA, Ora: *The Semantic Web : A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities*. URL http://www-personal.si.umich.edu/~rfrost/courses/SI110/readings/In_Out_and_Beyond/Semantic_Web.pdf – Überprüfungsdatum 2008-04-02

Bertram 2005

BERTRAM, Jutta: *Einführung in die inhaltliche Erschließung : Grundlagen, Methoden, Instrumente*. Würzburg : Ergon-Verl., 2005 (Content and communication 2). – ISBN 3899134427

Blanken u. a. 2007

BLANKEN, Henk M. ; BLOK, Henk Ernst ; FENG, Ling ; VRIES, Arjen P.: *Multimedia Retrieval*. Berlin, Heidelberg : Springer-Verlag Berlin Heidelberg, 2007 (Springer-11645 /Dig. Serial]). – ISBN 9783540728948

Bray u. a. 16.08.2006

BRAY, Tim; HOLLANDER, Dave; LAYMAN, Andrew; TOBIN, Richard: *Namespaces in XML 1.0*. URL <http://www.w3.org/TR/REC-xml-names/>. – Aktualisierungsdatum: 2006-08-16 – Überprüfungsdatum 2008-04-24

Cavazza, Green, Palmer 1998

CAVAZZA, Marc ; GREEN, Roger ; PALMER, Ian: *Multimedia Semantic Features and Image Content Description*. In: *Multimedia Modeling* (1998), 12-15, S. 39–46

Daconta, Obrst, Smith 2003

DACONTA, Michael C. ; OBRST, Leo J. ; SMITH, Kevin T.: *The semantic web: A guide to the future of XML, web services, and knowledge management*. Indianapolis, Ind. : Wiley, 2003 (Programming, software development). – ISBN 9780471432579

ebiquity research group 2007

EBIQUITY RESEARCH GROUP: *Swoogle Semantic Web Search Engine*. URL <http://swoogle.umbc.edu/> – Überprüfungsdatum 2008-04-12

Eckstein, Eckstein 2004

ECKSTEIN, Rainer ; ECKSTEIN, Silke: *XML und Datenmodellierung : XML-Schema und RDF zur Modellierung von Daten und Metadaten einsetzen*. 1. Aufl. Heidelberg : dpunkt-Verl., 2004. – ISBN 3898642224

Euzenat, Shvaiko 2007

EUZENAT, Jérôme ; SHVAIKO, Pavel: *Ontology matching*. Berlin, Heidelberg : Springer-Verlag Berlin Heidelberg, 2007 (Springer-11645 /Dig. Serial]). – ISBN 9783540496113

Freie Universität Berlin 2007

FREIE UNIVERSITÄT BERLIN: *Swoogle - VNBI - TWiki*. URL <https://www.inf.fu-berlin.de/w/VNBI/Swoogle>. – Aktualisierungsdatum: 2007-02-08 – Überprüfungsdatum 2008-04-12

Fukushige 2004

FUKUSHIGE, Yoshio: *Representing Probabilistic Knowledge in the Semantic Web*. URL <http://www.w3.org/2004/09/13-Yoshio/PositionPaper.html> – Überprüfungsdatum 2008-05-26

Google 12.07.2008

GOOGLE: *Unternehmensbezogene Informationen zu Google : Unternehmensprofil*. URL <http://www.google.com/intl/de/corporate/>. – Aktualisierungsdatum: 2008-07-12 – Überprüfungsdatum 2008-07-14

Handschuh, Staab 2003

HANDSCHUH, Siegfried ; STAAB, Steffen: *Annotation for the semantic web*. Amsterdam : IOS Press [u.a.], 2003 (Frontiers in artificial intelligence and applications 96). – ISBN 9781586033453

Hein 2008

HEIN, Markus: *pixelio.de : Deine kostenlose Bilddatenbank für lizenzfreie Fotos*. URL <http://www.pixelio.de/> – Überprüfungsdatum 2008-07-14

Herman 2008

HERMAN, Ivan: *W3C Semantic Web Activity*. URL <http://www.w3.org/2001/sw/>. – Aktualisierungsdatum: 2008-04-16 – Überprüfungsdatum 2008-04-16

Herman 2008

HERMAN, Ivan: *W3C Semantic Web FAQ*. URL <http://www.w3.org/2001/sw/SW-FAQ>. – Aktualisierungsdatum: 2008-02-19 – Überprüfungsdatum 2008-04-17

Hermes u. a. 2003

HERMES, Thorsten ; MIENE, Andrea ; IOANNIDIS, George ; HERZOG, Otthein: *Automatische Indexierung von multimedialen Daten*. In: *Info* 718 (2003), Nr. 2, S. 101–108

Hillman 2005

HILLMAN, Diane: *Using Dublin Core*. URL <http://dublincore.org/documents/2005/11/07/usageguide/>. – Aktualisierungsdatum: 2005-11-07 – Überprüfungsdatum 2008-04-25

Hitzler 2008

HITZLER, Pascal: *Semantic Web : Grundlagen*. Berlin : Springer-Verlag, 2008 (eXamen.press). – ISBN 3783540339939

Holzheid 2005

HOLZHEID, Gudula: *Dublin Core, SGML und die Zukunft der Erschließung am Beispiel einer Studie zur Optimierung des Dokumentenmanagements einer großen Nichtregierungsorganisation*. Berlin, 2005 (Berliner Handreichungen zur Bibliothekswissenschaft 136)

Hürst 2005

HÜRST, Wolfgang: *Multimediale Informationssuche in Vortrags- und Vorlesungsaufzeichnungen*. Dissertation. Freiburg, Albert-Ludwigs-Universität. Dissertation. 2005. – Aktualisierungsdatum: 2005

Hüsemann 2005

HÜSEMANN, Bodo: *Entwurf und Realisierung von Ontologien für Multimedia-Anwendungen*. Berlin : Akad. Verl.-Ges. Aka, 2005 (Dissertationen zu Datenbanken und Informationssystemen 89). – ISBN 3-89838-489-6

Isele 2005

ISELE, Christian: *Stand und Entwicklungsperspektiven des Semantischen Web*. Diplomarbeit. Hamburg, Hochschule für Angewandte Wissenschaften, Fb Bibliothek und Information. Diplomarbeit. 2005

Koivunen, Miller 2001

KOIVUNEN, Marja-Riitta; MILLER, Eric: *W3C Semantic Web Activity*
– Überprüfungsdatum 2008-04-12

Kompatsiaris, Hobson 2008

KOMPATSIARIS, Yiannis ; HOBSON, Paola: *Semantic Multimedia and Ontologies : Theory and Applications*. 1. Ed. Goldaming : Springer London, 2008. – ISBN 9781848000759

Kosch 2004

KOSCH, Harald: *Distributed multimedia database technologies supported by MPEG-7 and MPEG-21*. Boca Raton : CRC Press, 2004. – ISBN 0849318548

Kunz 2006

KUNZ, Christoph Daniel: *Ein integrierter Ansatz zur wissensbasierten Informationsrecherche*. Heimsheim, Stuttgart : Jost-Jetter; Univ., 2006 (IPA-IAO-Forschung und -Praxis 436). – ISBN 3-936947-85-6

Kuropka 2004

KUROPKA, Dominik: *Modelle zur Repräsentation natürlichsprachlicher Dokumente : Ontologie-basiertes Information-Filtering und -Retrieval mit relationalen Datenbanken*. Berlin : Logos-Verl., 2004 (Advances in information systems and management science 10). – ISBN 978-3-8325-0514-1

Lauriere 2007

LAURIERE, Stephane: *Welcome to SemanticDesktop.org*. URL <http://www.semanticdesktop.org/xwiki/bin/view/Main/>. – Aktualisierungsdatum: 2007-06-12 – Überprüfungsdatum 2008-04-12

LEO GmbH 2008

LEO GMBH: *WWWleo.org*. URL <http://www.leo.org/>. – Aktualisierungsdatum: 2008-07-14 – Überprüfungsdatum 2008-07-14

Liang u. a. 2006

LIANG, Anita C. ; LAUSER, Boris ; SINI, Margherita ; KEIZER, Johannes ; KATZ, Stephen: *From AGROVOC to the Agricultural Ontology Service Concept Server : Concept Server*. An OWL model for managing ontologies. 2006. – An OWL model for managing ontologies

Lux 2007

LUX, Mathias: *SemanticMetaddata.net : Caliph & Emir*. URL <http://www.semanticmetadata.net/features/>

Manjunath 2002

MANJUNATH, B. S.: *Introduction to MPEG-7 : Multimedia content description interface ; [DVD included]*. Chichester : Wiley, 2002. – ISBN 0471486787

Manola, Miller 2004

MANOLA, Frank; MILLER, Eric: *RDF Primer*. URL <http://www.w3.org/TR/REC-rdf-syntax/>. – Aktualisierungsdatum: 2004-02-10 – Überprüfungsdatum 2008-04-30

Miene, Hermes, Ioannidis 2002

MIENE, Andrea ; HERMES, Thorsten ; IOANNIDIS, George: *Wie kommt das Bild in die Datenbank? : Inhaltsbasierte Analyse von Bildern und Videos*. In: *Information Wissenschaft und Praxis* 53 (2002), S. 15–21

Munzinger Online 2007

MUNZINGER ONLINE: *Ontologie*. URL <http://www.munzinger.de/>. – Aktualisierungsdatum: 2007-07-01 – Überprüfungsdatum 2008-07-14

MySQL 2008

MYSQL: *MySQL 5.1 Referenzhandbuch : 12.7 MySQL-Volltextsuche*. URL <http://dev.mysql.com/doc/refman/5.1/de/fulltext-search.html> – Überprüfungsdatum 2008-08-30

Nack, Lindsay 1999

NACK, Frank ; LINDSAY, Adam T.: *Everything you wanted to know about MPEG-7*. In: *IEEE Multimedia* 6 (1999), Nr. 3, S. 65–77

Noy, McGuinness 2001

NOY, Natalya F. ; MCGUINNESS, Deborah L.: *Ontology Development 101 : A guide to creating your first ontology*. Stanford, 2001

Powers 2003

POWERS, Shelley: *Practical RDF : [solving problems with the resource description framework]*. 1. ed. Beijing : O'Reilly, 2003. – ISBN 0596002637

Quan 2007

QUAN, Dennis: *Improving life sciences information retrieval using semantic web technology*. In: *Briefings in Bioinformatics* 8 (2007), Nr. 3, S. 172–182. URL <http://bib.oxfordjournals.org/cgi/content/full/8/3/172?eaf> – Überprüfungsdatum 2008-04-02

Refnes Data 2008

REFNES DATA: *W3 Schools : RDF Containers*. URL http://www.w3schools.com/rdf/rdf_containers.asp – Überprüfungsdatum 2008-05-01

Richter 2008

RICHTER, Jörg: *DeepaMehta*. URL <http://www.deepamehta.de/>. – Aktualisierungsdatum: 2008-04-08 – Überprüfungsdatum 2008-04-12

Sachs 2006

SACHS, Eliza: *Getting Started with Protege-Frames*. 2006

Schaffert 2006

SCHAFFERT, Sebastian: *IkeWiki : A Semantic Wiki for Collaborative Knowledge Management*. In: *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2006. WETICE '06. 15th IEEE International Workshops on* (2006), S. 388–396

Schaffert 2007

SCHAFFERT, SEBASTIAN: *"Semantische Wikis haben hohen Nutzen als Werkzeug, um Lehre und Forschung besser zu organisieren"*. URL <http://www.semantic-web.at/index.php?id=1&subid=36&action=resource&item=166>. – Aktualisierungsdatum: 2007-03-19 – Überprüfungsdatum 2008-04-12

Schönbein 2006

SCHÖNBEIN, Rainer: *Agenten- und ontologiebasierte Software-Architektur zur interaktiven Bildauswertung*. Karlsruhe : Univ.-Verl. Karlsruhe, 2006. – ISBN 3-937300-98-8

Schulz 2008

SCHULZ, Ursula: *a-step : Volltextinvertierung*. Lektüre 1. URL http://www.bui.haw-hamburg.de/pers/ursula.schulz/astep/le6_step_3.html. – Aktualisierungsdatum: 2008-04-20 – Überprüfungsdatum 2008-05-14

Shadbolt, Hall, Berners-Lee 2006

SHADBOLT, Nigel ; HALL, Wendy ; BERNERS-LEE, Tim: *The Semantic Web Revisited*. In: *IEEE Intelligent Systems* 21 (2006), Nr. 3, S. 96–101. URL www.computer.org/intelligent

Sizov 2007

SIZOV, Sergej: *What Makes You Think That? The Semantic Web's Proof Layer*. In: *IEEE Intelligent Systems* 22 (2007), Nr. 6, S. 94–99

Smith, Welty, McGuiness 12.04.2006

SMITH, Michael K.; WELTY, Chris; MCGUINNESS, Deborah L.: *OWL Web Ontology Language Guide : Deutsche Übersetzung*. URL <http://www.semaweb.org/dokumente/w3/TR/2004/REC-owl-guide-20040210-DE.html>. – Aktualisierungsdatum: 2006-04-12 – Überprüfungsdatum 2008-05-05

Spiegel ONLINE 2008

SPIEGEL ONLINE: *Das Internet soll klüger werden*. In: *Spiegel ONLINE* (2008). URL <http://www.spiegel.de/netzwelt/web/0,1518,561831,00.html>

Staab 2004

STAAB, Steffen: *Handbook on ontologies : With 22 tables*. Berlin : Springer, 2004 (International handbooks on information systems). – ISBN 3540408347

Stamou, Kollias 2005

STAMOU, Giorgos ; KOLLIAS, Stefanos: *Multimedia content and the semantic web : Methods, standards and tools*. Chichester : Wiley, 2005. – ISBN 0470857536

Stanford Center for Biomedical Informatics Research 2008

STANFORD CENTER FOR BIOMEDICAL INFORMATICS RESEARCH: *The Protégé Ontology Editor and Knowledge Acquisition System*. URL <http://protege.stanford.edu/> – Überprüfungsdatum 2008-07-15

Stock 2007

STOCK, Wolfgang G.: *Information retrieval : Informationen suchen und finden ; [Lehrbuch]*. München : Oldenbourg, 2007 (Einführung in die Informationswissenschaft 1). – ISBN 3486581724

Sullivan 2008

SULLIVAN, Nicole: *The Friend of a Friend (FOAF) project*. URL <http://www.foaf-project.org/index.html> – Überprüfungsdatum 2008-04-13

Tarabanis 2006

TARABANIS, Konstantinos: *SemanticGov: Services for Public Administration*. URL <http://www.semantic-gov.org/index.php> –
Überprüfungsdatum 2008-04-13

Umlauf 2000

UMLAUF, Konrad: *Inhaltserschließung in Bibliotheken*. Berlin, 2000(Berliner Handreichungen zur Bibliothekswissenschaft 82)

W3C 2005 - Haystack

W3C: *Haystack: BioDASH Demonstration*. URL
<http://www.w3.org/2005/04/swls/BioDash/Demo/>. –
Aktualisierungsdatum: 2005-11-22 – Überprüfungsdatum 2008-04-02

W3C 2005 - What is Haystack?

W3C: *What is Haystack?* URL
<http://www.w3.org/2005/04/swls/BioDash/Demo/What%20is%20Haystack.html>. – Aktualisierungsdatum: 2005-05-08 –
Überprüfungsdatum 2008-07-14

Wagner 2006

WAGNER, Karl Heinz: *Darstellung von Wissen*. URL
<http://www.fb10.uni-bremen.de/khwagner/semantik/wissen.htm>. –
Aktualisierungsdatum: 2006-10-22 – Überprüfungsdatum 2008-07-14

Werner 1997

WERNER, Walter: *Definitionen und Einordnung*. URL
<http://santana.uni-muenster.de/Linguistik/user/steiner/semindex/intro.html>. –
Aktualisierungsdatum: 1997-05-30 – Überprüfungsdatum 2008-07-14

Wikipedia 2008 - Agricultural Ontology Service

WIKIPEDIA: *Agricultural Ontology Service*. URL
http://de.wikipedia.org/wiki/Agricultural_Ontology_Service. –
Aktualisierungsdatum: 2008-06-09 – Überprüfungsdatum 2008-07-14

Wikipedia 2008 - Formale Semantik

WIKIPEDIA: *Formale Semantik*. URL
http://de.wikipedia.org/wiki/Formale_Semantik. –
Aktualisierungsdatum: 2008-07-11 – Überprüfungsdatum 2008-07-14

Wikipedia 2008 - Information-Retrieval

WIKIPEDIA: *Information-Retrieval*. URL
http://de.wikipedia.org/wiki/Information_retrieval. –
Aktualisierungsdatum: 2008-07-13 – Überprüfungsdatum 2008-07-14

Wikipedia 2008 - Ontologie (Informatik)

WIKIPEDIA: *Ontologie (Informatik)*. URL
http://de.wikipedia.org/wiki/Ontologie_%28Informatik%29. –
Aktualisierungsdatum: 2008-07-12 – Überprüfungsdatum 2008-07-14

Wikipedia 2008 - Sacherschließung

WIKIPEDIA: *Sacherschließung*. URL
<http://de.wikipedia.org/wiki/Inhaltserschlie%C3%9Fung>. –
Aktualisierungsdatum: 2008-06-22 – Überprüfungsdatum 2008-07-14

Wikipedia 2008 - Semantik

WIKIPEDIA: *Semantik*. URL <http://de.wikipedia.org/wiki/Semantik>. –
Aktualisierungsdatum: 2008-07-08 – Überprüfungsdatum 2008-07-14

Wikipedia 2008 - Semantisches Web

WIKIPEDIA: *Semantisches Web*. URL
http://de.wikipedia.org/wiki/Semantisches_Web. –
Aktualisierungsdatum: 2008-06-30 – Überprüfungsdatum 2008-07-14

Wikipedia 2008 - YouTube

WIKIPEDIA: *YouTube*. URL <http://de.wikipedia.org/wiki/Youtube>. –
Aktualisierungsdatum: 2008-07-12 – Überprüfungsdatum 2008-07-14

Yu 2007

YU, Liyang: *Introduction to the Semantic Web and Semantic Web services*. Boca Raton, Fla. : Chapman & Hall/CRC, 2007. – ISBN 1584889330

Anhang

Einfache Ontologie – Multimediaontologie.owl

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns="http://www.owl-
ontologies.com/MultimediaBeispielOntologie.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.owl-
ontologies.com/MultimediaBeispielOntologie.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Visuelle_Metadaten_Video">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Formale_Metadaten_Video"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Formale_Metadaten_Video">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Metadaten"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Video">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Thematische_Metadaten_Video"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Thematische_Metadaten_Video">
    <rdfs:subClassOf rdf:resource="#Visuelle_Metadaten_Video"/>
  </owl:Class>
  <owl:Class rdf:ID="Foto">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Thematische_Metadaten_Foto"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Formale_Metadaten_Foto">
    <rdfs:subClassOf rdf:resource="#Metadaten"/>
  </owl:Class>
  <owl:Class rdf:ID="Visuelle_Metadaten_Foto">
    <rdfs:subClassOf rdf:resource="#Formale_Metadaten_Foto"/>
  </owl:Class>
```

```

<owl:Class rdf:about="#Thematische_Metadaten_Foto">
  <rdfs:subClassOf rdf:resource="#Visuelle_Metadaten_Foto"/>
</owl:Class>
<owl:DatatypeProperty rdf:ID="Schlagworte">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Thematische_Metadaten_Foto"/>
        <owl:Class rdf:about="#Thematische_Metadaten_Video"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Begriff">
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Systematik">
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Thematische_Metadaten_Foto"/>
          <owl:Class rdf:about="#Thematische_Metadaten_Video"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Album">
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
          <owl:Class rdf:about="#Formale_Metadaten_Video"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
  </owl:DatatypeProperty>
  <owl:FunctionalProperty rdf:ID="Mpeg7_Texture">
    <rdfs:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
    <rdfs:domain>

```

```

<owl:Class>
  <owl:unionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#Visuelle_Metadaten_Foto"/>
    <owl:Class rdf:about="#Visuelle_Metadaten_Video"/>
  </owl:unionOf>
</owl:Class>
</rdfs:domain>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="Dateiname">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
        <owl:Class rdf:about="#Formale_Metadaten_Video"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
    <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Dateiname mit jeweiliger Dateiendung</rdfs:comment>
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:FunctionalProperty>
  <owl:FunctionalProperty rdf:ID="Dokumenttitel">
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
      <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
        <owl:Class rdf:about="#Formale_Metadaten_Video"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</owl:FunctionalProperty>
  <owl:FunctionalProperty rdf:ID="Kameramodell">
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
    <rdfs:domain>
    <owl:Class>

```

```

    <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
      <owl:Class rdf:about="#Formale_Metadaten_Video"/>
    </owl:unionOf>
  </owl:Class>
</rdfs:domain>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="Mpeg7_Motion">
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:domain rdf:resource="#Visuelle_Metadaten_Video"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="Erstellungsdatum">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
        <owl:Class rdf:about="#Formale_Metadaten_Video"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Erstellungsdatum mit Wochentag und Datum</rdfs:comment>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="Dateiumfang">
  <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Dokumentumfang in MB</rdfs:comment>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
        <owl:Class rdf:about="#Formale_Metadaten_Video"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
</owl:FunctionalProperty>

```

```

<owl:FunctionalProperty rdf:ID="Urheber">
  <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Urheber mit Vor- und Nachnamen</rdfs:comment>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain>
  <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
      <owl:Class rdf:about="#Formale_Metadaten_Video"/>
    </owl:unionOf>
  </owl:Class>
</rdfs:domain>
<rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="Mpeg7_Color">
  <rdfs:domain rdf:resource="#Visuelle_Metadaten_Foto"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<Foto rdf:ID="Foto_1"/>
<Video rdf:ID="Video1">
  <Dokumenttitel
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Testfilm</Dokumenttitel>
  <Schlagworte xml:lang="de">Wohnung</Schlagworte>
  <Schlagworte xml:lang="de">Kurzfilm</Schlagworte>
  <Mpeg7_Texture
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
  >554478884</Mpeg7_Texture>
  <Album xml:lang="de">Testalbum</Album>
  <Dateiumfang
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
  >22.25</Dateiumfang>
  <Mpeg7_Motion
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
  >224545454</Mpeg7_Motion>
  <Dateiname rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Test.avi</Dateiname>
  <rdfs:comment xml:lang="de">Videobeschreibung</rdfs:comment>
  <Urheber rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Peter Schindler</Urheber>
  <Schlagworte xml:lang="de">Test</Schlagworte>
  <Album xml:lang="de">Videos von zu Hause</Album>

```

```

    <Erstellungsdatum
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Montag 30.06.2008</Erstellungsdatum>
    <Systematik xml:lang="de">Privat</Systematik>
</Video>
<Video rdf:ID="Video2">
    <Dateiname rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Test2.avi</Dateiname>
    <Dateiumfang
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
    >19.84</Dateiumfang>
    <Schlagworte xml:lang="de">Test</Schlagworte>
    <Album xml:lang="de">Testalbum</Album>
    <Erstellungsdatum
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Montag 30.06.2008</Erstellungsdatum>
    <Mpeg7_Motion
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >154541212</Mpeg7_Motion>
    <Album xml:lang="de">Videos aus der Arbeit</Album>
    <Systematik xml:lang="de">Privat</Systematik>
    <Systematik xml:lang="de">Arbeit</Systematik>
    <Mpeg7_Texture
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >232185645</Mpeg7_Texture>
    <Dokumenttitel
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Testfilm2</Dokumenttitel>
    <Urheber rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Peter Schindler</Urheber>
    <Schlagworte xml:lang="de">Arbeitsplatz</Schlagworte>
    <Schlagworte xml:lang="de">Kurzfilm</Schlagworte>
</Video>
</rdf:RDF>

```

Komplexe Ontologie – Politik.owl

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns="http://www.owl-
ontologies.com/MultimediaBeispielOntologie.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.owl-
ontologies.com/MultimediaBeispielOntologie.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Visuelle_Metadaten_Video">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Formale_Metadaten_Video"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Formale_Metadaten_Video">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Metadaten"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Video">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Thematische_Metadaten_Video"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Thematische_Metadaten_Video">
    <rdfs:subClassOf rdf:resource="#Visuelle_Metadaten_Video"/>
  </owl:Class>
  <owl:Class rdf:ID="Foto">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Thematische_Metadaten_Foto"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Formale_Metadaten_Foto">
    <rdfs:subClassOf rdf:resource="#Metadaten"/>
  </owl:Class>
  <owl:Class rdf:ID="Visuelle_Metadaten_Foto">
    <rdfs:subClassOf rdf:resource="#Formale_Metadaten_Foto"/>
  </owl:Class>
  <owl:Class rdf:about="#Thematische_Metadaten_Foto">
    <rdfs:subClassOf rdf:resource="#Visuelle_Metadaten_Foto"/>
  </owl:Class>
  <owl:DatatypeProperty rdf:ID="Schlagworte">
```

```

<rdfs:domain>
  <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Thematische_Metadaten_Foto"/>
      <owl:Class rdf:about="#Thematische_Metadaten_Video"/>
    </owl:unionOf>
  </owl:Class>
</rdfs:domain>
<rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="Begriff">
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="Systematik">
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</rdfs:domain>
  <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Thematische_Metadaten_Foto"/>
      <owl:Class rdf:about="#Thematische_Metadaten_Video"/>
    </owl:unionOf>
  </owl:Class>
</rdfs:domain>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="Album">
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</rdfs:domain>
  <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
      <owl:Class rdf:about="#Formale_Metadaten_Video"/>
    </owl:unionOf>
  </owl:Class>
</rdfs:domain>
</owl:DatatypeProperty>
<owl:FunctionalProperty rdf:ID="Mpeg7_Texture">
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Visuelle_Metadaten_Foto"/>
        <owl:Class rdf:about="#Visuelle_Metadaten_Video"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>

```

```

        </owl:unionOf>
    </owl:Class>
</rdfs:domain>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="Dateiname">
    <rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
                <owl:Class rdf:about="#Formale_Metadaten_Video"/>
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
        <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
        >Dateiname mit jeweiliger Dateiendung</rdfs:comment>
        <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:FunctionalProperty>
    <owl:FunctionalProperty rdf:ID="Dokumenttitel">
        <rdfs:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
        <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
                <owl:Class rdf:about="#Formale_Metadaten_Video"/>
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
</owl:FunctionalProperty>
    <owl:FunctionalProperty rdf:ID="Kameramodell">
        <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </rdfs:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
    </rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
                <owl:Class rdf:about="#Formale_Metadaten_Video"/>
            </owl:unionOf>

```

```

    </owl:Class>
  </rdfs:domain>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="Mpeg7_Motion">
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:domain rdf:resource="#Visuelle_Metadaten_Video"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="Erstellungsdatum">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
        <owl:Class rdf:about="#Formale_Metadaten_Video"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Erstellungsdatum mit Wochentag und Datum</rdfs:comment>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:FunctionalProperty>
  <owl:FunctionalProperty rdf:ID="Dateiumfang">
    <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Dokumentumfang in MB</rdfs:comment>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
          <owl:Class rdf:about="#Formale_Metadaten_Video"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
  </owl:FunctionalProperty>
  <owl:FunctionalProperty rdf:ID="Urheber">
    <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Urheber mit Vor- und Nachnamen</rdfs:comment>

```

```

<rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
<rdfs:domain>
  <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Formale_Metadaten_Foto"/>
      <owl:Class rdf:about="#Formale_Metadaten_Video"/>
    </owl:unionOf>
  </owl:Class>
</rdfs:domain>
<rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="Mpeg7_Color">
  <rdfs:domain rdf:resource="#Visuelle_Metadaten_Foto"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<Foto rdf:ID="Foto_1"/>
<Video rdf:ID="Video1">
  <Dokumenttitel
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Testfilm</Dokumenttitel>
  <Schlagworte xml:lang="de">Wohnung</Schlagworte>
  <Schlagworte xml:lang="de">Kurzfilm</Schlagworte>
  <Mpeg7_Texture
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
  >554478884</Mpeg7_Texture>
  <Album xml:lang="de">Testalbum</Album>
  <Dateiumfang
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
  >22.25</Dateiumfang>
  <Mpeg7_Motion
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
  >224545454</Mpeg7_Motion>
  <Dateiname rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Test.avi</Dateiname>
  <rdfs:comment xml:lang="de">Videobeschreibung</rdfs:comment>
  <Urheber rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Peter Schindler</Urheber>
  <Schlagworte xml:lang="de">Test</Schlagworte>
  <Album xml:lang="de">Videos von zu Hause</Album>
  <Erstellungsdatum
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Montag 30.06.2008</Erstellungsdatum>
  <Systematik xml:lang="de">Privat</Systematik>

```

```

</Video>
<Video rdf:ID="Video2">
  <Dateiname rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Test2.avi</Dateiname>
  <Dateiumfang
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
  >19.84</Dateiumfang>
  <Schlagworte xml:lang="de">Test</Schlagworte>
  <Album xml:lang="de">Testalbum</Album>
  <Erstellungsdatum
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Montag 30.06.2008</Erstellungsdatum>
  <Mpeg7_Motion
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
  >154541212</Mpeg7_Motion>
  <Album xml:lang="de">Videos aus der Arbeit</Album>
  <Systematik xml:lang="de">Privat</Systematik>
  <Systematik xml:lang="de">Arbeit</Systematik>
  <Mpeg7_Texture
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
  >232185645</Mpeg7_Texture>
  <Dokumenttitel
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Testfilm2</Dokumenttitel>
  <Urheber rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Peter Schindler</Urheber>
  <Schlagworte xml:lang="de">Arbeitsplatz</Schlagworte>
  <Schlagworte xml:lang="de">Kurzfilm</Schlagworte>
</Video>
</rdf:RDF>

```

Eidstattliche Versicherung

Ich versichere, die vorliegende Arbeit selbstständig ohne Hilfe verfasst und keine anderen Quellen und Hilfsmittel als die angegebenen benutzt zu haben. Die aus anderen Werken wörtlich entnommenen Stellen oder dem Sinn nach entlehnten Passagen sind durch Quellenangabe kenntlich gemacht.

Hamburg, 18.08.2008 _____