

Universität Koblenz-Landau

Seminar Semantic Grid

WS 04/05

Prof. Dr. Steffen Staab

Betreuung des Seminars:

Bernhard Tausch

Seminararbeit

Thema II.1

Basis des Semantic Web:

Das Resource Description Framework (RDF)

Von

Katrin Frank

Matrikelnummer 20 111 00 11

katrin.frank@uni-koblenz.de

Seminar-Vortrag: 12.01.2005

Inhaltsverzeichnis

BASIS DES SEMANTIC WEB: DAS RESOURCE DESCRIPTION FRAMEWORK (RDF)	1
INHALTSVERZEICHNIS	2
ABBILDUNGSVERZEICHNIS	3
1 EINFÜHRUNG	4
2 DAS SEMANTIC WEB	4
2.1 DIE IDEE DES SEMANTIC WEB	5
2.2 ARCHITEKTUR DES SEMANTIK WEB	8
2.2.1 <i>Unicode</i>	9
2.2.2 <i>URI</i>	9
2.2.3 <i>XML, Namespace</i>	9
2.2.4 <i>RDF</i>	10
2.2.5 <i>Ontologie</i>	10
2.2.6 <i>Web of Trust</i>	11
2.3 ANWENDUNGSBEREICHE	11
2.3.1 <i>E-Commerce</i>	11
2.3.2 <i>Suchmaschinen</i>	11
2.3.4 <i>Softwarespezifikation</i>	12
2.3.5 <i>Personenbezogene Daten</i>	12
3 RDF	13
3.1 RDF-MODELL	13
3.2. NOTATIONEN	14
3.2.1 <i>Tripel Notation</i>	14
3.2.2 <i>Gerichteter Graph</i>	15
3.2.3 <i>RDF/XML</i>	15
3.3 RDF-SCHEMA (RDFS)	16
3.3.1 <i>Klassenkonzept</i>	16
3.3.2 <i>Klassen</i>	16
3.3.3 <i>Eigenschaften</i>	17
3.4 CONTAINER	18
3.4.1 <i>Bag</i>	18
3.4.2 <i>Sequence</i>	19
3.4.3 <i>Alternative</i>	19
3.5 COLLECTIONS	20
3.6 RDF-ANWENDUNGEN	21
3.6.1 <i>Dublin Core</i>	21
3.6.2 <i>PhotoProject</i>	21
3.6.3 <i>Sesame</i>	22
4 KRITIK	23
5 AUSBLICK	24
LITERATURVERZEICHNIS	25

Abbildungsverzeichnis

ABBILDUNG 1: TIM BERNERS-LEE	5
ABBILDUNG 2: BISHERIGE WEB-STRUKTUR	7
ABBILDUNG 3: SEMANTIC WEB-STRUKTUR.....	7
ABBILDUNG 4: SEMANTIC WEB, STICHWORT "WEBER".....	8
ABBILDUNG 5: LAYER-CAKE VON TIM BERNERS-LEE	8
ABBILDUNG 6: WEB OF TRUST	11
ABBILDUNG 7: RDF TRIPEL.....	13
ABBILDUNG 8: RDF DESCRIPTION.....	13
ABBILDUNG 9: RDF DESCRIPTION BEISPIEL.....	13
ABBILDUNG 10: GERICHTETER GRAPH	15
ABBILDUNG 11: ABHÄNGIGKEITEN DER RDF KLASSEN UNTEREINANDER.....	17
ABBILDUNG 12: BAG BEISPIEL	19
ABBILDUNG 13: SEQUENCE BEISPIEL.....	19
ABBILDUNG 14: ALTERNATIVE BEISPIEL	19
ABBILDUNG 15: RDF COLLECTION BEISPIEL.....	20
ABBILDUNG 16: PHOTOPROJECT ARCHITEKTUR.....	22
ABBILDUNG 17: SESAME ARCHITEKTUR.....	23

1 Einführung

Die folgende Ausarbeitung befasst sich mit der Basis des Semantic Web, einem Teilbereich des zum Wintersemester angebotenen Seminars „Semantic Grid“, gelehrt von Prof. Dr. Staab und betreut von Bernhard Tausch. Im Speziellen soll dabei das Resource Description Framework vorgestellt werden.

Die Ausarbeitung teilt sich in drei Abschnitte auf. Zu Beginn werde ich das Semantic Web vorstellen, wie es dazu kam, welche Ideen dahinter stecken und wo das Semantic Web Anwendung findet. Der zweite Teil befasst sich dann spezieller mit dem Resource Description Framework, kurz RDF, dem RDF-Modell und seinem Schema. Den Schluss bilden eine Kritik und einige Informationen zur künftigen Aussicht des RDFs und des Semantic Webs.

2 Das Semantic Web

Wie kam es zum Semantic Web? Warum ist es entstanden? Um diese Frage zu beantworten, muss man sich mit der aktuellen Situation des Internets beschäftigen. Mittlerweile ist das World Wide Web zum weltgrößten Datenspeicher heran gewachsen und wird sich in Zukunft bezüglich enthaltener Daten sicherlich noch vergrößern. Der weltweite Zugriff auf Informationen verbessert sich täglich. Das Internet ist damit eine riesige Sammlung von Dokumenten, Dateien und Informationen, die weltweit publiziert werden. Ein Ideal wäre es, irgendwann das gesamte Weltwissen der Menschheit hierdurch repräsentieren zu können. Ein Wunschvorstellung, die zu erreichen aber auch einige Probleme mitbringt. Schon jetzt fasst das Web so viele Daten, dass sich die Frage stellt, wie man unter dieser riesigen Informationsflut an spezielle Daten heran kommt. Kommt man überhaupt an sie heran oder bleiben einige Quellen sogar verschlossen? Und ist es genau die Information, die man suchte? Schon jetzt gleicht das Suchen nach speziellen Informationen im Internet einer Suche der Stecknadel im Heuhaufen.

Ein Beispiel: Aufgabe sei, eine 20 Seiten Ausarbeitung über das Thema Deadlink zu erstellen. Eine verzweifelte Suche nach Information beginnt. Für Literatur ist dieses Thema zu jung, Suchmaschinen müssen gefüttert werden. Fakt ist jedoch, wissenschaftliche Texte oder lexikalische Erläuterungen sind unter den ersten 100 Treffern so gut wie nicht zu finden. Die meisten Treffer bestehen aus kurzen Hinweisen bzw. der Bitte an User der Webseiten, falls dieser einen Deadlink entdeckt haben sollte, eine Nachricht an den Betreiber der Seite zu senden. Nach dieser größten Treffermenge der Deadlink-Melde-Aufforderungen, findet die Suchmaschinen Verlinkungen zu einigen angebotenen Programmen, die Deadlinks selbstständig überprüfen und falls nötig entfernen. Erklärungen sind jedoch so gut wie unaufindbar.

Ein weiteres einfaches Beispiel: Eine Sexualpsychologin sucht einen wissenschaftlichen Text über das Erotik-Verhalten von Mann und Frau in der Frühlingszeit. Wie viel wissenschaftliche Ausarbeitungen mögen hier unter den ersten 100 Treffern wohl enthalten sein?

Worin liegt diese Unzulänglichkeit der derzeit zur Verfügung stehenden Suchmöglichkeiten bei Suchmaschinen? Leider erlauben diese nur das Auffinden von Seiten durch den Vergleich von Zeichenketten. Die Bedeutung des eigentlichen Inhalts bleibt der Maschine verschlossen und kann nicht als Suchkriterium verwendet werden. Den gefunden Link auszuwerten, ist Aufgabe des Menschen und diese Aufgabe erweist sich als nicht gerade klein.

Die maschinenlesbare Information auch in ihrer semantischen Bedeutung den Maschinen verständlich machen zu können, könnte zu einer erheblichen Verbesserung bei Suche von Information führen. Ein Beispiel:

```
<h1> Name: Hans Mueller </h1>  
<h2> Straße: Bahnhofsgasse 3 </h2>
```

Dem menschlichen Anwender ist beim gemeinsamen Betrachten der beiden Zeilen sofort klar, dass es sich hier um eine Adresse handelt. Die Maschine erkennt das nicht, da sie auf rein syntaktischer Basis arbeitet.

Gerade im Hinblick auf die enorm steigende Anzahl an Dokumentendaten, wird es in Zukunft immer wichtiger werden, diese Daten in einer Art und Weise zu beschreiben, die nicht nur der Mensch, sondern auch die Maschine verarbeiten, auswerten und verstehen kann.

Agenten, so genannte eigenständige Programme, könnten gezielt eingesetzt werden und auch Suchmaschinen würden qualitativ hochwertigere Ergebnisse erzielen. Das Semantic Web fasst diese Idee auf und versucht genau das durch ein einheitliches Beschreibungsmodell zu ermöglichen.

2.1 Die Idee des Semantic Web

Das Konzept des Semantic Web wurde entwickelt von der Institution „World Wide Web Consortium“, auch bekannt unter der Abkürzung W3C. Das W3C 1994 entstand in Zusammenarbeit mit dem Europäischen Kernforschungszentrum CERN in Genf (Schweiz) am Massachusetts Institute of Technology (MIT) in Cambridge (USA).



Abbildung 1: Tim Berners-Lee

Unter dem Vorsitz des Briten Sir Timothy J. Berners-Lee, der auch schon die ersten Ideen zum World Wide Web (WWW) lieferte, forschen heute über 400 Mitglieder, darunter zahlreiche Forschungseinrichtungen und Industrievertreter, an der ständigen Weiterentwicklung des WWW und damit auch an dem Semantic Web. Für Berners-Lee's Beitrag zur Wissenschaft, den er durch die Erfindung des World Wide Web leistete, wurde er letztes Jahr (16. Juli 2004) durch Königin Elisabeth II. zum Knight Commander, Order of the British Empire (KBE), zum Ritter geschlagen.¹

Das W3C arbeitet außerdem an speziellen Dateiformaten, Repräsentation, Architektur und Systemen wie XHTML, XML, XML-Schema, Voice-XML etc., die ebenfalls für das Semantic Web maßgeblich sind. Die Abteilung Technology & Society des W3C nimmt sich politischen Problemen an und versucht für diese Lösungen zu entwickeln.

Die Problematik, mit der sich das W3C bezüglich Semantic Web beschäftigt, besteht darin Technologien und Standards für ein universelles, flexibles und maschinenverständliches Datenformat zu schaffen, das Maschinen das Auswerten von Informationen, ermöglicht.

Wie kann man die im Internet vorhandenen Informationen für Maschinen verständlich machen? Dafür gibt es 2 Ansätze:

1) Maschinen können dazu gebracht werden natürliche Sprache zu verstehen. Forschungen im Bereich Computerlinguistik liefern allerdings keine allgemeine und zufrieden stellende Lösung. Neueste Erkenntnisse der Computerlinguistik vermuten, dass es einer Maschine niemals möglich sein wird, die menschliche Sprache vor allem die gesprochene Sprache voll-ständig verstehen zu können.

2) Eine andere Möglichkeit ist es Daten in einer Art und Weise aufzubereiten, dass sie von Maschinen verstanden werden können. Das bedeutet einerseits einen Mehraufwand für die Informationsaufbereitung und -Bereitstellung, vermeidet aber auch die oben erläuterten Probleme weitestgehend, da die Computer (Maschinen) dann sozusagen in ihrer Sprache kommunizieren können. Dieser Ansatz wird im Semantic Web verfolgt.²

Anmerken muss man, dass eine vollständige Auswertung aller Informationen einer Repräsentation laut den Erkenntnissen von Gödel nie erreicht werden kann. Nach diesem Satz ist es nicht möglich, alle richtigen Theoreme und Erkenntnisse eines Axiomen-Systems mittels gegebener Kalküle abzuleiten und zu beweisen. So beschränken sich die Forschung damit zumindest den größtmöglichen Teil der Daten im WWW für Anwendungen zugänglich zu machen.³

¹ Vgl. Ct Magazin Ausgabe 06/2002

² Vgl. W3C Semantic Web

³ Vgl. W3C The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities by Tim Berners-Lee, James Hendler and Ora Lassila

Realisiert wird dies mit so genannten Metadaten. Dies sind Informationen über Informationen, in unserem Fall maschinenverarbeitbare Informationen über Webinhalte. Diese Daten über Daten, sind für den User unsichtbar mit dem Code der Website verknüpft. Hierauf basieren sämtliche Semantic Web-Anwendungen. Will man diese weltweit nutzen, ist eine Standard-Sprache für die Beschreibung dieser Metadaten notwendig. Diese muss sowohl den Datenaustausch unabhängig von spezieller Software, als auch die Datenverarbeitung vereinfachen, semantische Beziehungen zwischen Ressourcen beschreiben und ein Verstehen der Information auf semantischer Ebene ermöglichen. Maschinen sollen selbständig Daten im Internet verwerten können.

Das Semantic Web soll das bisherige Web nicht ersetzen, sondern auf ihm aufbauen. Die Sprache XML (Extensible Markup Language) liefert den Grundbaustein. Das RDF liefert die Syntax, die Ontologie die Zusammenhänge, beschrieben durch die RDF-Syntax, so dass das System dadurch in der Lage ist logische Schlussfolgerungen zu ziehen.

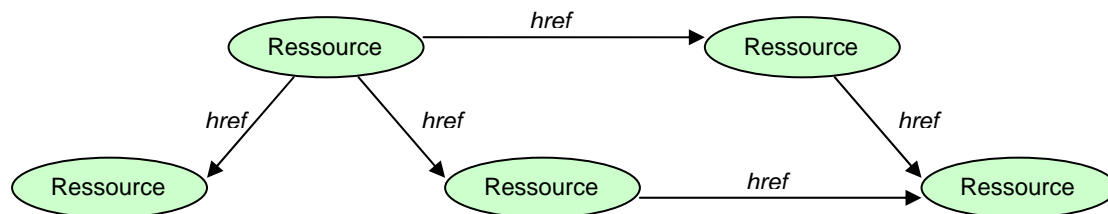


Abbildung 2: Bisherige Web-Struktur

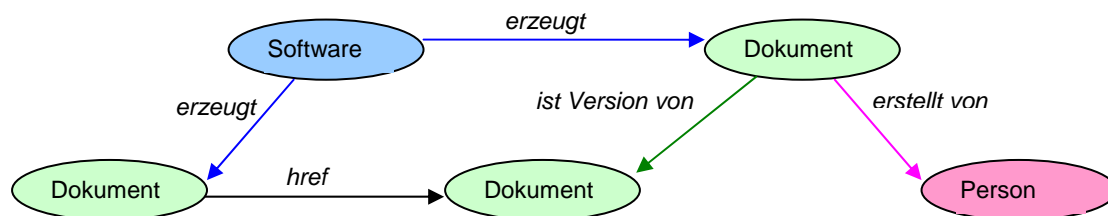


Abbildung 3: Semantic Web-Struktur

Für einen konkreten Fall könnte das wie folgt aussehen: Zu der Zeichenkette „Weber“ könnte man wie in der unten gezeigten Grafik z.B. folgende Zusammenhänge und Einträge erhalten.

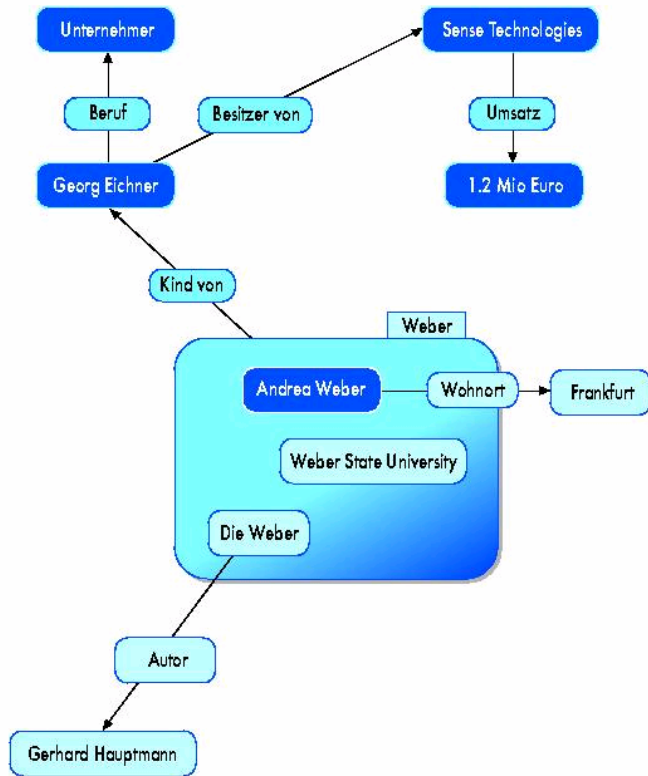


Abbildung 4: Semantic Web, Stichwort "Weber"

2.2 Architektur des Semantic Web

Damit Maschinen selbstständig nutzbringende Informationen herausfiltern und verwerten können, ist eine hierarchische Struktur von Semantic Web Sprachen und verschiedenen Mechanismen notwendig. Wie wir schon wissen, baut das Semantic Web auf dem bisherigen Web auf. Dies bedeutet, es muss mit anderen Webtechnologien kompatibel sein. Die folgende Grafik „Layer-cake“ von Berners-Lee beschreibt die Architektur des Semantic Web.

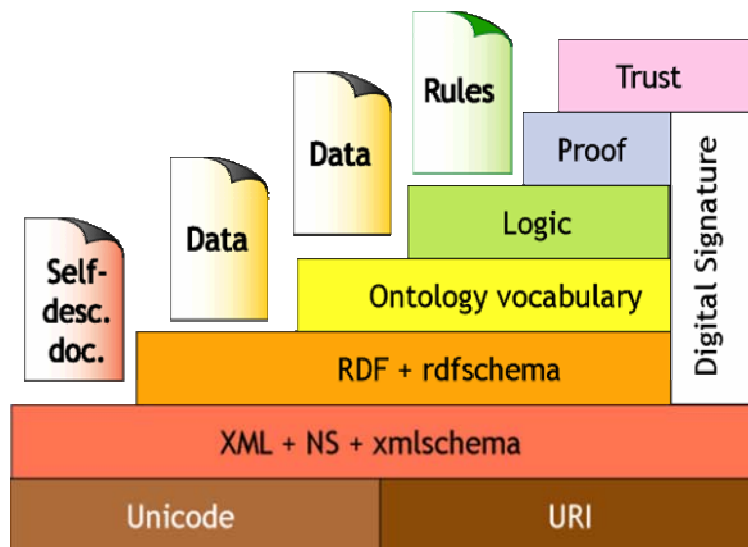


Abbildung 5: Layer-cake von Tim Berners-Lee

2.2.1 Unicode

Bei dem Unicode handelt es sich um einen weltweit universellen Zeichensatz. Diese ISO Norm 10646 stellt sicher, dass international einheitliche Standards zur Bezeichnung im World Wide Web verwendet werden.

2.2.2 URI

Der „Uniform Resource Identifier“, kurz URI genannt, wurde bereits zu Beginn des World Wide Web eingeführt und als eindeutiger Bezeichner für eine Ressource im Netz vom W3C festgelegt. Um über ein bestimmtes Objekt zu sprechen, muss dieses eindeutig identifizierbar sein, dies wird durch die URI ermöglicht. Jede Ressource im Internet hat ihre einmalige URI, sie weist einen eindeutigen Namen und einen eindeutigen Pfad zu.

Eine bekannte Unter-Form der URI ist die URL („Uniform Resource Locator“). Der Unterschied besteht darin, dass die URL der Ressource einen eindeutigen Namen vergibt, und sie auch gleichzeitig lokalisiert, so z.B. bei der URL von Webseiten (www.hastdu-nichtgesehen.de). Die URI dagegen muss keinen Pfad zu dieser Ressource aufzeigen.

2.2.3 XML, Namespace

Während HTML die Daten visuell wiedergibt, wird XML (Extensible Markup Language) von Programmen zum Datenaustausch im Netz verwendet. Hierzu ist es sinnvoll der Maschine mehr Informationen über ihren Dateninhalt zu geben, z.B.:

```
<sentence>
  <person href="http://www.katrin.de">Ich</person>
  habe einen
  <animal>Hund</animal>.
</sentence>
```

Diese Mehrinformationen wären in vorliegendem Fall, dass es sich insgesamt um einen Satz handelt, dass „Ich“ eine Person ist und dass „Hund“ ein Tier ist.

Um Problemen der Mehrdeutigkeiten vorzubeugen, werden dem so genannten Namespace (Namensraum) Präfixe vorangestellt. Im folgenden Beispiel kommt <news> doppelt vor:

```
<cnn:news>...</cnn:news> und <n24:news>...</n24:news>
```

Die gleichen Namespaces mit URI sehen wie folgt aus:

```
<cnn:news xmlns:cnn="http://www.cnn.com/xmlns/">...</cnn:news>
<n24:news xmlns:n24="http://www.n24.de/">...</n24:news>
```

Wichtig und für einen Erfolg notwendig ist, dass das plattformunabhängige XML als W3C-Technologie lizenzfrei ist und von der Industrie unterstützt wird.

2.2.4 RDF

Um verarbeitbare Aussagen über Ressourcen machen können, ist XML nicht mehr ausreichend. Das von W3C entwickelte so genannte „Resource Description Framework“ baut auf XML auf und bietet erweiterte einheitliche Beschreibungsmöglichkeiten für den semantischen Inhalt von Daten. Genauer gesagt, RDF ist ein Modell zur Repräsentation von Metadaten. Der Vorteil ist, dass jede RDF-Aussage aus einem einfachen Tripel besteht, das sich aus Subjekt, Prädikate und Objekt zusammensetzt. Hierauf wird in Kapitel 3 der Ausarbeitung noch näher eingegangen. Durch RDF alleine lassen sich jedoch noch keine semantischen Verknüpfungen erstellen. Dazu wird eine weitere Schicht benötigt, die Ontologie-Schicht.

2.2.5 Ontologie

Die Ontologie ist eine formale Beschreibung von Gegenständen, vor allem deren Beziehungen und Verbindungen zu einander. Sie basiert auf der RDF-Syntax und dem RDF-Schema, stellt darüber hinaus aber noch weitere Sprachkonstrukte auf, die sehr der Sprache Prolog ähneln. Stellt man sich vor, RDF sei die Programmiersprache, so wäre die Ontologie das Programm, das damit geschrieben wird.

Es könnte z.B. sein, dass die Bedeutung des Inhalts zweier Artikel der Gleiche ist, die URI's sind bekanntlich jedoch unterschiedlich. Die Ontologie kann hier als Träger von den Informationen eingesetzt werden, die einer Maschine klar machen kann, dass es inhaltlich um das Gleiche geht.

Noch ein Beispiel:

PKW und LKW gehören beide zu dem Überbegriff Fahrzeug und bilden Spezialisierungen bzw. Unterklassen. Ein Agent sucht im bisherigen Netz nach einem Fahrzeug mit dem Baujahr 1977. In seiner Suche stößt er auf einen PKW mit dem Baujahr 1977. Ohne das Wissen, dass ein PKW ein Fahrzeug ist, wird dieser Fund wieder verworfen.

Es existiert nicht „die eine“ Ontologie, sondern verschiedene Vertreter davon. Dazu gehören z.B.: OIL (Ontology Inference Layer), eine Entwicklung der Europäer und die parallele Entwicklung DAML der Amerikaner. Da sich beide Forschungen in dieselbe Richtung bewegten, einigte man sich und schloss sich zusammen. Aus DAML+OIL wurde eine gemeinsame Initiative, die Ontologiesprache OWL (Web Ontologie Language).

„Übrigens, das Akronym für *Web Ontology Language* hätte eigentlich *WOL*, nicht *OWL* sein müssen. In Referenz auf die literarische Figur der Eule aus Milnes *Pu der Bär*, die als einziges Tier im Wald ihren Namen schreiben konnte (und denselben aber mit einem Buchstabendreher im englischen Original *WOL* statt *OWL* schreibt), wurde dieser Buchstabendreher umgekehrt für die OWL übernommen.“⁴

⁴ Vgl. Wikipedia: Ontologie Language OWL.

Die oberen Stufen des „Layer-cake“, Logik, Proof und Trust befinden sich noch in der Entwicklung. Oberstes Ziel soll ein Web des Vertrauens sein.

2.2.6 Web of Trust

Berners-Lee's Wunschvorstellung ist ein „Web of Trust“. Es zielt auf das Vertrauen ab, dass man seinen Freunden entgegenbringt. Nach dem Motto: „Der Freund meines Freundes ist auch mein Freund!“, geht es in dem Web of Trust um unterschiedliche Grade der Vertrauensstufen.

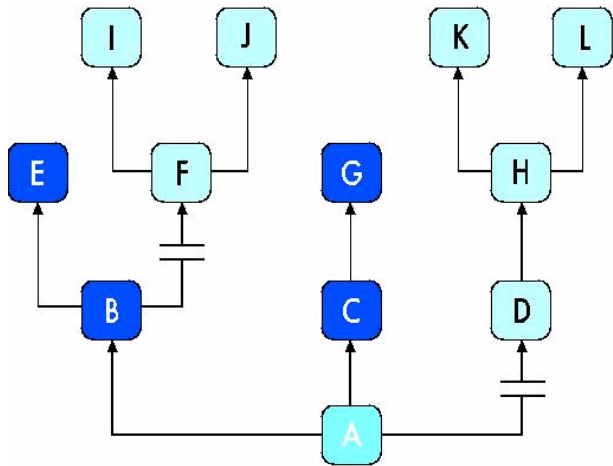


Abbildung 6: Web of Trust

Dem dunkelblauen Knoten B und C vertraut die Instanz A, zu ihnen führen offene Verbindungen, B vertraut wiederum E, C vertraut G, also vertraut A insgesamt B, C, E, G. Den restlichen Knoten vertraut A nicht, zu ihnen sind die Leitungen gesperrt. Genauso wie man jemanden Vertrauen entgegenbringt, ist es ebenso möglich, einer Instanz (Person) das Vertrauen abzusprechen.

2.3 Anwendungsbereiche

Die neue Form von Informationen des Semantic Web macht auch eine Reihe von neuen Anwendungsideen möglich.

2.3.1 E-Commerce

Gerade der Bereich „Vertrauen“ ist beim E-Commerce ein wichtiger Aspekt. Hier bietet das Semantic Web weit aus mehr Möglichkeiten als bisher XML.

2.3.2 Suchmaschinen

Die aktuellen Suchmaschinen arbeiten mit Wortvergleichen. Dies bedeutet, es werden alle Dokumente als Treffer ausgegeben, in denen das gesuchte Wort vorkommt, unabhängig von der Sprache oder Bedeutung. Oft ist nach einer Suche die Ergebnismenge enorm groß, aber das, was man sucht, ist nicht dabei. Das Semantic Web könnte die Suche erheblich verbessern: z.B. Suche Weber als Beruf, nicht den Namen.

2.3.3 Agenten

Ein weiteres Anwendungsgebiet sind so genannte „Agenten“. Sie können sich selbstständig im Internet bewegen und verschiedene Aufgaben abwickeln, z.B. die Suche nach einem passenden Arzt, Urlaubsort oder bestimmten Termin. Für die Entwicklung eines solchen Programms stellt das Semantic Web sogar eine Voraussetzung dar, allerdings reicht es dazu alleine nicht vollkommen aus. Es sind noch weitere Voraussetzungen nötig.

2.3.4 Softwarespezifikation

Ist z.B. die Syntax eines Programmcodes nicht bekannt, ist es auch nicht möglich die Semantik dahinter zu verstehen. Dies gilt für Maschine und Mensch in gleicher Weise. Noch existieren keine Programme, die der Maschine das Lesen der Bedeutung unbekannter Syntax ermöglichen.

Ein Forschungsbereich der KI (Künstlichen Intelligenz) ist seit einiger Zeit damit beschäftigt, die Semantik unabhängig von einer Syntax zu definieren. Ziel ist es, dass auch der Anwender mit Hilfe geeigneter Software Informationen besser versteht, ohne spezielle Syntaxkenntnisse (Programmierkenntnisse) zu besitzen.

2.3.5 Personenbezogene Daten

Informationen über Personen könnten in den Metadaten abgelegt werden, was eine Art Kontrolle ermöglicht. Personendaten könnten leichter auffindig gemacht und Profile könnten erstellt werden. Ein großer Vorteil ergäbe sich in dieser Weise für das E-Commerce, dass auf Vertrauen der Teilnehmer basiert. Außerdem könnte es zur Bekämpfung der Piraterie im Internet beitragen, kriminelle Webseiten aufdecken und der Ermittlung von Verantwortlichen dienen.

Große Kritik muss man aussprechen aufgrund der Tatsache, dass genauso auch Falschinformationen über Personen verbreitet werden können. Viel wichtiger ist hier aber die Frage des Datenschutzes.

3 RDF

Das Resource Description Framework wurde vom World Wide Web Consortium (W3C) entwickelt und bietet eine Grundlage für den Metadaten-Austausch. Ein großes Problem dieses Austausches stellt die Vielzahl unterschiedlicher Datenformate dar. RDF bietet hierfür eine einfache Syntax und ein Schema an und ermöglicht so eine einheitliche Beschreibung.

3.1 RDF-Modell

Ein Vorteil von RDF liegt in der Einfachheit seiner Syntax. Wie ein einfacher Satz unserer Sprache, besteht in RDF jede Aussage (engl. Description) aus drei Komponenten, den so genannten „Tripels“ oder „Statements“: Subjekt, Prädikat und Objekt.



Abbildung 7: RDF Tripel

Eine Verknüpfung des Tripels zur semantischen Beschreibung zeigt dieser Graph. Dabei sind Subjekt und Objekt die Knoten und das Prädikat die Kante. Die Ressource (Subjekt) wird durch eine Menge von *Properties* (Eigenschaften) beschrieben, was bedeutet, dass vom Subjekt durchaus mehrere Kante ausgehen können. *Properties* bestehen jeweils aus einem *Property-Typ* (engl.: type; hier Prädikat) und einem *Property-Wert* (engl.: value; hier Objekt). Die Ressource und die Menge aller *Properties* ergeben zusammen die *RDF-Description* (auf dt.: RDF-Beschreibung).

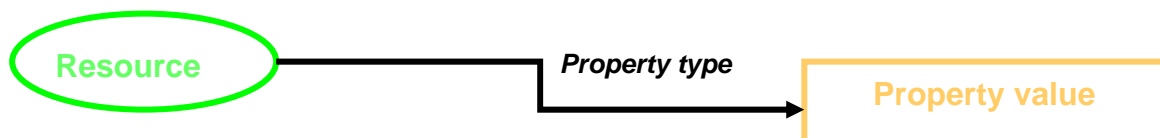


Abbildung 8: RDF Description

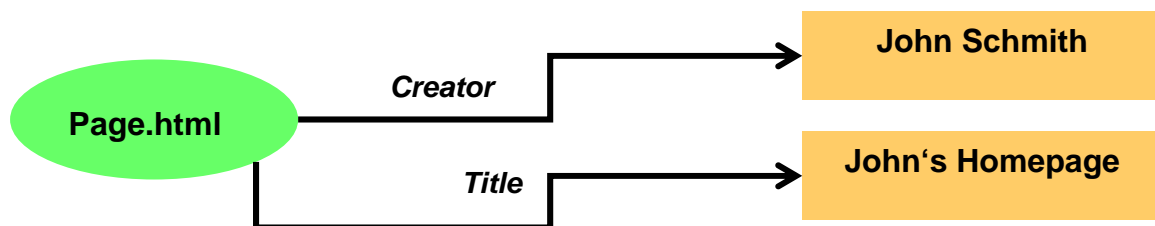


Abbildung 9: RDF Description Beispiel

Alle Bestandteile einer *Description* sind jeweils eindeutig durch eine URI identifizierbar. Ein Objekt kann auch ein Literal sein, das heißt aus Ziffern oder Buchstaben bestehen. Subjekt, Prädikat und Objekt können wiederum selbst Ressourcen für ein anderes Subjekt, Prädikat und Objekt darstellen. Dies ermöglicht eine Verschachtelung und ist unter „Reification“ bekannt.

3.2. Notationen

RDF besitzt mehrere Darstellungsmöglichkeiten. Die wichtigsten sind nachfolgend aufgeführt: Tripel Notation, Gerichteter Graph und RDF/XML-Notation. Sie alle beschreiben den Beispielsatz:

„*http://www.example.org/index.html* has a *creationdate* whose value is *August 16, 1999*.“

3.2.1 Tripel Notation

http://www.example.org/index.html has a **creationdate** whose value is **August 16, 1999**

Subjekt	http://www.example.org/index.html
Prädikat	http://www.example.org/terms/creation-date
Objekt	“August 16, 1999”

Das Subjekt ist die URL „http://www.example.org/index.html“, Prädikat ist das Wort „creationdate“ (Erstellungsdatum) und Objekt ist Phrase „August 16, 1999“.

Wie schon erwähnt, wird allen drei Komponenten eine eindeutige URI zugewiesen, und das Objekt kann ebenfalls ein Literal sein, wie es bei „August 16,1999“ der Fall ist. In dieser Notation werden alle drei Bestandteile hintereinander bzw. untereinander aufgelistet.

Eine Tripel Darstellung mit drei URIs ohne Literal könnte wie folgt aussehen:

http://www.example.org/index.html has a **creator** whose value is **John Smith**.

Subjekt	http://www.example.org/index.html
Prädikat	http://purl.org/dc/elements/1.1/creator
Objekt	http://www.example.org/staffid/85740

Dem Objekt John Smith wird in diesem Beispiel eine ID zugeordnet.

3.2.2 Gerichteter Graph

<http://www.example.org/index.html> has a **creationdate** whose value is **August 16, 1999**

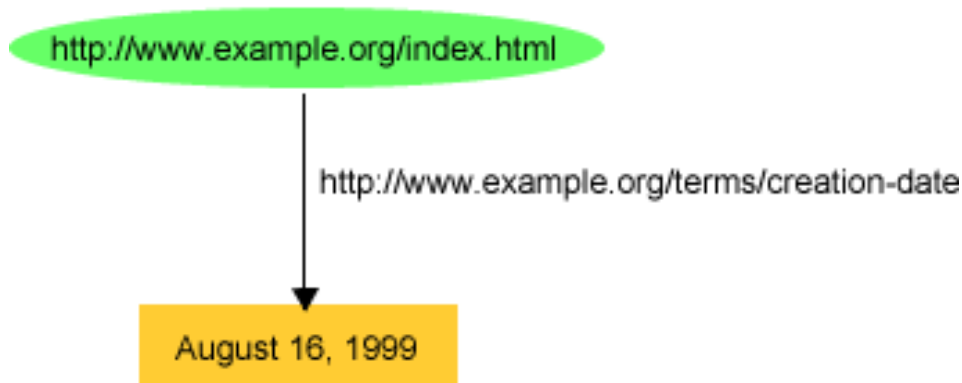


Abbildung 10: Gerichteter Graph

In gerichteten Graphen werden Ressource (Subjekt) und Objekt normalerweise in grünen Ellipsen dargestellt. Handelt es sich bei dem Objekt um Literale, wird es wie hier in einem orangenen Rechtecken abgebildet. Die URI des Prädikates bildet die Kantenbeschriftung.

3.2.3 RDF/XML

RDF/XML ist ebenfalls eine W3C Spezifikation und wird offiziell von diesem empfohlen. Auf der Basis der Sprache XML, werden RDF-Schemas (siehe Kapitel 3.3) beschrieben. RDF/XML kann unsichtbar direkt in den HTML-Code der Webseite integriert werden.

<http://www.example.org/index.html> has a **creationdate** whose value is **August 16, 1999**

```

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:externs="http://www.example.org/terms/">

  <rdf:Description rdf:about="http://www.example.org/index.html">
    <externs:creation-date>
      August 16, 1999
    </externs:creation-date>
  </rdf:Description>
</rdf:RDF>
  
```

Die erste Zeile zeigt an, dass der folgende Code vom Typ XML der Version 1.0 ist. Zuerst wird in XML/RDF mit `xmlns` der Namespace definiert und `externs` zur Verfügung gestellt. Die eigentliche Beschreibung beginnt darunter. Mit `Description.. about` wird festgelegt, über welche Ressource eine Aussage getroffen wird. Entsprechend wird die Eigenschaft `creation-date` mit dem Wert `August 16, 1999` dieser Ressource zugeordnet.

3.3 RDF-Schema (RDFS)

Das RDF-Modell mit seinen drei Syntax-Komponenten bietet keine Möglichkeit Eigenschaften und Beziehungen zu beschreiben. Abhilfe schafft eine Erweiterung zum so genannten RDF-Schema. Mit dessen Hilfe können nun Zusammenhänge beschrieben und Einschränkungen des Wertebereiches gemacht werden. Als besonders nützlich erweist sich ähnlich einer objektorientierten Programmiersprache das Klassenprinzip.

3.3.1 Klassenkonzept

Das Klassenkonzept ermöglicht durch seinen hierarchischen Aufbau Vererbung, Mehrfachvererbung, Spezialisierung und Einschränkung von einzelnen Klassen und einzelnen Eigenschaften. Es können Oberklassen und Unterklassen (Instanzen dieser Klasse), Obereigenschaften und Untereigenschaften gebildet werden. Im Nachfolgenden werden einige der wichtigsten Klassen und Eigenschaften und ihre jeweilige Bedeutung näher beschrieben.

3.3.2 Klassen

rdfs:Resource

- bezeichnet alles, was mit RDF beschrieben werden kann;
- ist oberste Klasse, alle anderen Klassen sind Instanzen von ihr;

rdfs:Class

- ist Klasse aller Klassen;
- dient in Verbindung mit *rdf:type* zur Erzeugung von Instanzen;

rdfs:Property

- ist die Basisklasse für Eigenschaften, d.h. Elemente, die zur Beschreibung von Ressourcen verwendet werden;

rdfs:Literal

- ist Klasse für alle Literale, also Zeichenketten, Integer etc.;

rdf:type

- gibt an, ob eine Ressource Instanz einer Klasse ist;
- Wertebereiche von *rdf:type* sind Instanzen von *rdfs:Class*;

3.3.3 Eigenschaften

rdfs:type

- identifiziert die Klasse einer Ressource;

rdfs:subClassOf

- legt die Vererbungshierarchien von Klassen fest;
- Hierarchiebildung ähnlich der objektorientierten Programmierung;

rdfs:subPropertyOf

- legt eine Instanz von *rdf:Property* an;
- gibt an, dass eine Eigenschaft eine nähere Beschreibung einer anderen Eigenschaft ist;

rdfs:domain

- legt den Anwendungsbereich einer Eigenschaft in Bezug auf eine Klasse fest;
- z.B. der Eigenschaft „hatFarbe“ kann sowohl der Klasse Blume als auch der Klasse Ball zugeordnet werden;

rdfs:range

- legt den Wertebereich einer Eigenschaft fest;
- z.B. die Eigenschaft „istFanVonHelgeSchneider“ kann die Werte TRUE oder FALSE annehmen;

rdfs:seeAlso

- gibt eine Ressource an, die zusätzliche Information über die zu beschreibende Ressource beinhaltet;

rdfs:comment

- fügt Kommentare hinzu

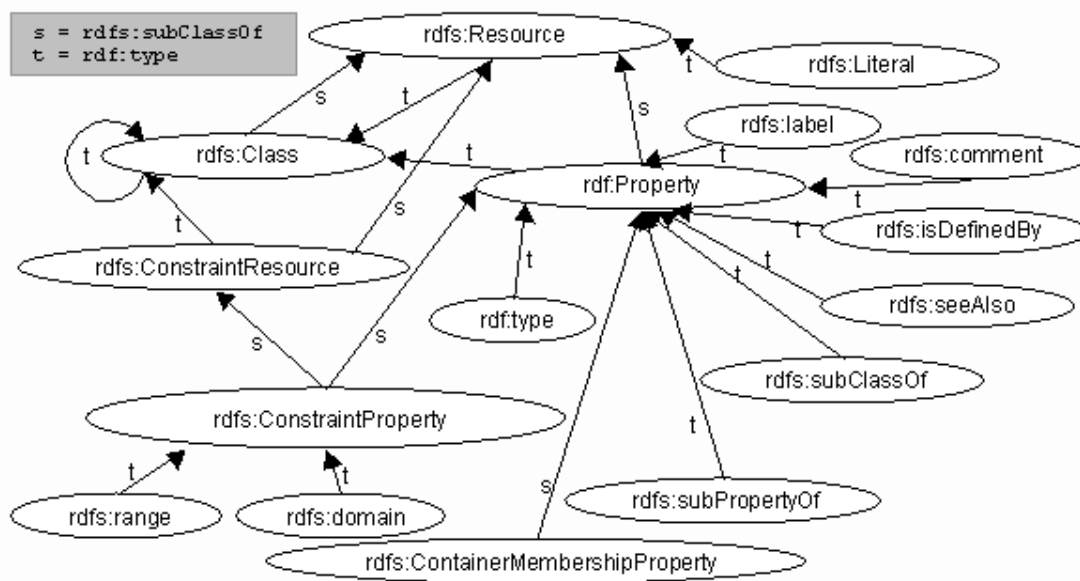


Abbildung 11: Abhängigkeiten der RDF Klassen untereinander

Ein Beispiel für eine Vererbung durch `subClassOf` zeigt der nachfolgende Code.

```
<rdfs:Class rdf:ID="MotorVehicle">
  <rdfs:subClassOf
    rdf:resource="http://www.w3.org/2000/01/rdf-
      schema#Resource"/>
</rdfs:Class>

<rdfs:Class rdf:ID="Van">
  <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
</rdf:Class >

<rdfs:Class rdf:ID="MiniVan">
  <rdfs:subClassOf rdf:resource="#Van"/>
  <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
</rdf:Class>
```

In der ersten Klasse wird die Ressource „MotorVehicle“ (dt.: motorisiertes Fahrzeug) beschrieben. Sie ist Unterklasse von der höchsten Klasse `rdfs:Resource` (siehe Abb. 11). Die zweite Klasse „Van“ ist eine Spezialisierung von „Motor Vehicle“, somit Unterklasse von dieser und erbt alle Eigenschaften von „MotorVehicle“. Die letzte Klasse „Mini Van“ ist eine Spezialisierung von „Van“, was bedeutet, dass sie Unterklasse von „Van“, aber auch Unterklasse von „Motor Vehicle“ ist und natürlich auch von beiden die spezifischen Eigenschaften erbt.

3.4 Container

Der Container besteht aus drei verschiedenen Ressourcen-Typen:

- *rdf:Bag* (Ungeordnete Menge von Ressourcen)
- *rdf:Seq* (Geordnete Menge von Ressourcen)
- *rdf:Alt* (Menge von alternativen Ressourcen)

3.4.1 Bag

Bag bezeichnet eine unsortierte Liste von Ressourcen oder Literalen ohne Hierarchie, das heißt es ist keine Reihenfolge der Gruppenmitglieder festgelegt. Doppelte Mitglieder sind erlaubt. Ein *Bag* könnte z.B. genutzt werden, um eine Gruppe von Teilnehmern zu beschreiben, in der die Reihenfolge von Prozess-Beginnen vollkommen egal ist.

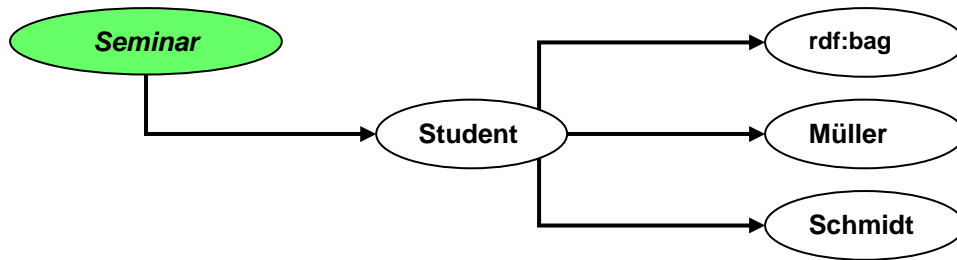


Abbildung 12: Bag Beispiel

3.4.2 Sequence

Sequence bezeichnet eine sortierte Liste von Ressourcen oder Literalen mit Hierarchie, das heißt mit festgelegter Reihenfolge. Doppelte Mitglieder sind auch hier erlaubt. Eine *Sequence* könnte z.B. genutzt werden, um eine Gruppe von Teilnehmern zu beschreiben, die alphabetische Reihenfolge einhalten muss.

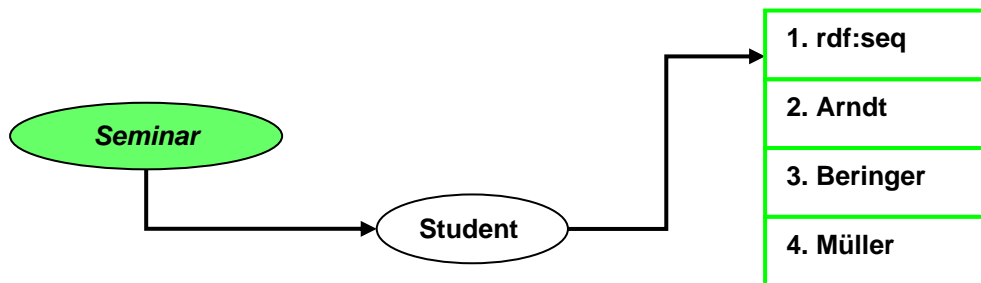


Abbildung 13: Sequence Beispiel

3.4.3 Alternative

Alternative bezeichnet eine unsortierte Liste von Ressourcen oder Literalen ohne Hierarchie, die allerdings nur alternativ gewählt werden dürfen (typisch für einzelne Werte von Eigenschaften). Ein *Alternative* könnte z.B. genutzt werden, um alternative Übersetzungssprachen für eine Website auszuwählen. Es ist das Anzeigen entweder in englisch, in deutsch oder in französisch möglich.

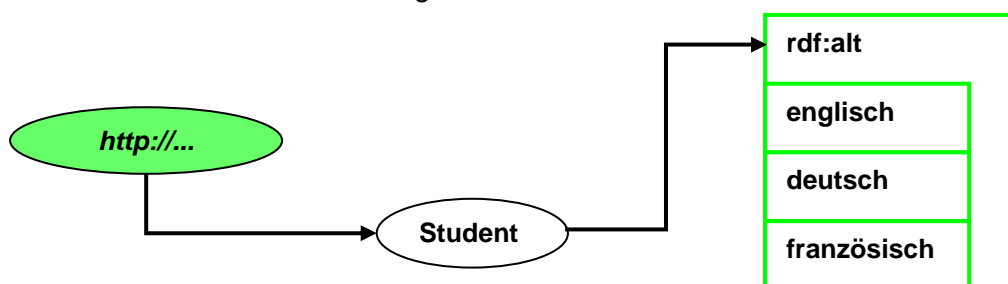


Abbildung 14: Alternative Beispiel

3.5 Collections

Ein Nachteil des Containers ist, er kann nicht explizit sagen: „Dies sind meine kompletten Inhalte.“ Er kann erkennen, dass es sich um ein Mitglied handelt, nicht aber, ob darüber hinaus noch weitere Mitglieder existieren.

RDF unterstützt die Beschreibung einer Gruppe von zusammengehörenden Mitgliedern in Form einer RDF Collection (Sammlung). Dargestellt in einer Graphen-Liste, nutzt diese Listenstruktur ein vordefiniertes Collection-Vokabular, bestehend aus:

- den vordefinierten Typen: *rdf:list*
- den vordefinierten Eigenschaften: *rdf:first* und *rdf:rest*
- der vordefinierten Ressource: *rdf:nil*

Veranschaulicht dargestellt ist dies im folgenden Graphen zum Satz:
„The students in course 6.001 are Amy, Mohamed and Johann“

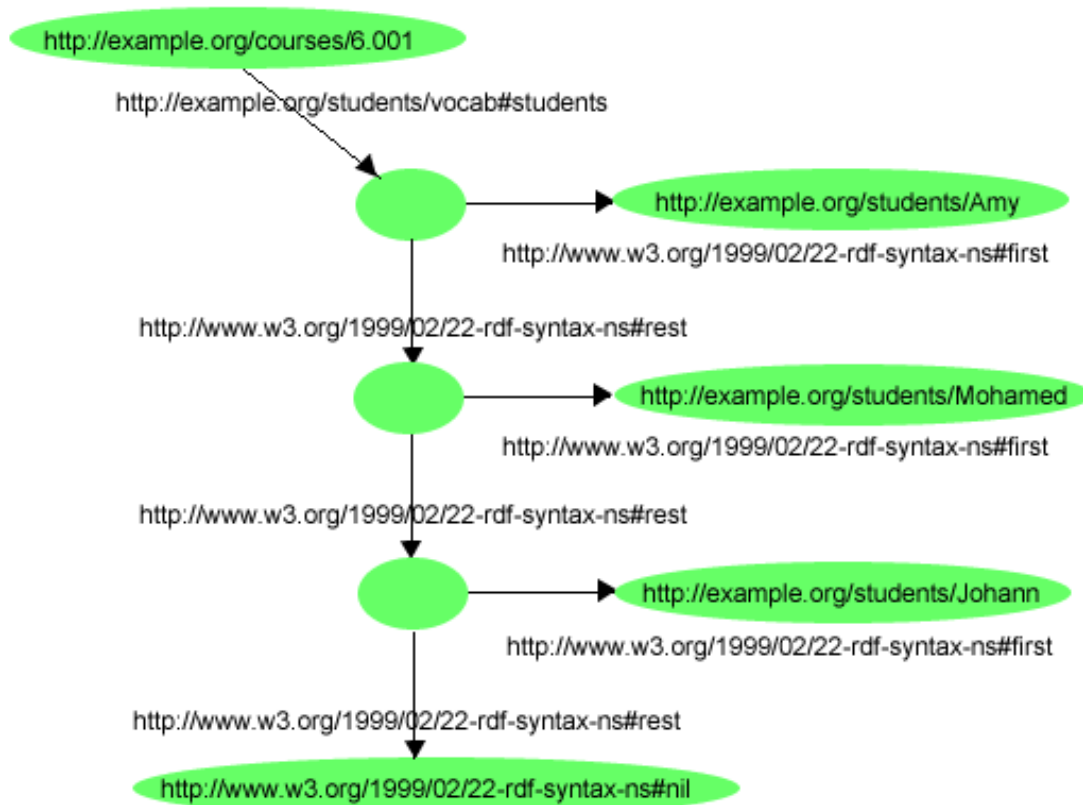


Abbildung 15: RDF Collection Beispiel

In diesem Graph ist jedes Mitglied der Collection (Sammlung bzw. Liste) wie z.B. *s:Amy*, das Objekt der Eigenschaft *rdf:first property*, dessen Subjekt in dem Fall eine leere Ressource ist, die wiederum die Liste repräsentiert. Diese Listen-Ressource ist verlinkt zur restlichen Liste mit Hilfe der Eigenschaft *rdf:rest property*.

Das Ende der Liste ist schließlich wieder zuerst gekennzeichnet mit der Eigenschaft *rdf:rest property*, die aber dann auf *rdf:nil* zeigt. Dieses *rdf:nil* stellt eine leere Ressource dar. Die Struktur ähnelt dem Prinzip der Programmiersprache Lisp.

3.6 RDF-Anwendungen

Im Nachfolgenden sollen Dublin Core, PhotoProject und Sesame als Vertreter von RDF Anwendung etwas näher erklärt werden. Neben diesen dreien existieren natürlich noch weit aus mehr Anwendungen, denen RDF zu Grund liegt, wie z.B.: PRISM, XPackage, RSS, CIM/XML, Gene Ontology Consortium, Annotea, FOAF(Friend of a Friend), UK Mirror Service, PICS oder P3P.

3.6.1 Dublin Core

Dublin Core besteht aus einem Set von Elementen (Eigenschaften) um Dokumente und deren Metadaten zu beschreiben. Das Set wurde im März 1995 in Dublin, Ohio entwickelt. Ziel ist es, ein minimales Set von Eigenschaften anzubieten. Es soll Beschreibungen, Indizierungen und die Retrieval Suche von dokumentenähnlichen Objekten im Netz erleichtert (ähnlich Bibliotheken-Kataloge). Außerdem unterstützt Dublin Core hierdurch Suchmaschinen weltweit.

Dublin Core ist ausreichend einfach zu verstehen und zu verwenden, um von einer Großzahl der Autoren und gelegentlichen Verlegern, die Information im Internet bereitstellen, genutzt werden zu können. Die aktuellen Elemente des Dublin Core sind im Dublin Core Metadata Element Set, Version 1.1: Reference Description [DC] definiert und bestehen aus dem folgenden Eigenschaften-Set:

*Title, Creator, Subject, Description, Publisher, Contributor, Date, Type, Format, Identifier, Source, Language, Relation, Coverage, Rights.*⁵

3.6.2 PhotoProject

Entstanden ist dieses Projekt einerseits aus persönlichem Interesse, andererseits um die W3C Technology zu fördern. Die persönlichen Gründe waren, dass Autoren des W3C immer wieder mit der Schwierigkeit des Auffindens eines bestimmten Fotos innerhalb einer großen Menge von Fotos zu kämpfen hatten. Die Digitalisierung des Fotos und eine Beschreibung dessen mit RDF könnte es für Suchen schneller auffindbar machen.

⁵ Vgl. W3C RDF Primer: Applications: Dublin Core.

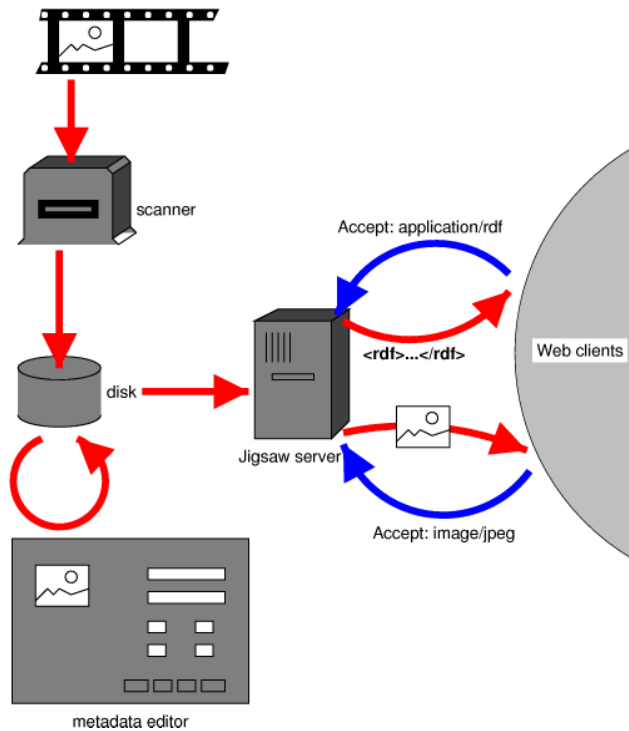


Abbildung 16: PhotoProject Architektur

Oben links: Die Bilder werden digitalisiert und als JPEG Format gespeichert.

Unten links: Metadaten werden in die Bilder mit dem „Data-entry Programm“ integriert (können auch nachträglich editiert werden, falls Korrekturen nötig sind).

Rechts: Anfragen vom Web werden mit Jigsaw gespeichert, indem entweder die Bilder oder die Metadaten, abhängig von der Form der Anfrage, versendet werden.⁶

3.6.3 Sesame

Die Zusammenarbeit von der Vrije Universiteit Amsterdam VUA (coordinator, NL), University of Karlsruhe (Germany), Schweizerische Lebensversicherungs- und Rentenanstalt/Swiss Life (Switzerland), British Telecommunications plc (UK), CognIT a.s. (Norway), EnerSearch AB (Sweden) und Administrator Nederland BV (NL) brachte ein Projekt names Sesame hervor, das allgemeine semantische Anfragen an eine Datenbank ermöglicht. Sesame bietet dies mit einer ersten Implementationen der Sprache RQL (Query Language) an.

⁶ Vgl. W3C: "Describing and retrieving photos using RDF and HTTP" Note, 28. September 2000.

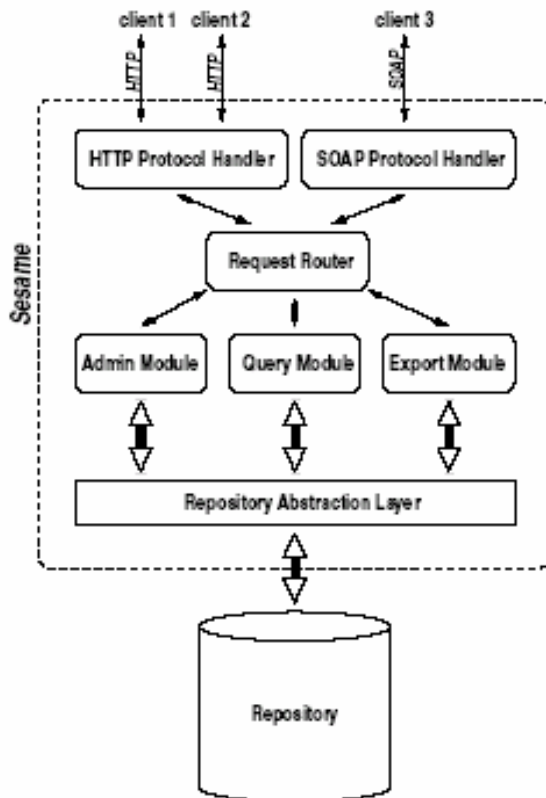


Abbildung 17: Sesame Architektur

4 Kritik

Für einen erfahrenen Internet-User, der sich auch mit der Erstellung von Webseiten mit Hilfe des HTML-Codes beschäftigt, ist der Schwierigkeitsgrad von HTML akzeptabel. Dies ist einer der Gründe, der zur enormen Ausbreitung des Internet führte. Ganz anders empfindet der Durchschnittsanwender, dieser ist bereits mit Erstellung von HTML-Seiten insbesondere HTML-Code überfordert. Die RDF-Syntax zur Beschreibung von Metadaten ist noch komplexer und weit aus schwieriger zu bewältigen als die Syntax des HTML-Codes. Denkbar wäre hier eine Art Editor, eine Anwendung, wie es sie schon zum Erstellen von HTML-Code gibt (z.B. Dreamweaver, Frontpage etc.). Diese könnte auch einem Laien das Hinzufügen von Metadaten mit Hilfe einer entsprechenden Oberfläche ermöglichen.

Eine positive Eigenschaft des HTML-Codes ist außerdem die Tatsache, dass ein unmittelbares visuelles Ergebnis sofort sichtbar wird. Das Hinzufügen von RDF bringt erst einmal keine direkt sichtbaren Vorteile. Das ist problematisch, da hierdurch die Notwendigkeit für den User nicht offensichtlich ist. So wird es wahrscheinlich erst einmal den Experten überlassen, das Internet mit RDF-Metadaten auszustatten, und eine schnelle weltweite Ausbreitung scheint unwahrscheinlich. Um aber Anwendungen, wie sie in dem Kapitel 2 beschrieben wurden, einsetzen zu können, müssten weltweit Webseiten mit RDF-Metadaten vorhanden sein.

Ein weiterer Kritikpunkt ist die Tatsache, dass sowohl das Semantic Web, als auch RDF/OWL beides Ideen und Konzepte des W3C sind. Hierdurch werden leider andere ähnliche Technologien mangels Unterstützung kaum beachtet. Eine dieser Technologien ist z.B.: XML Topic Maps. Dieses Konzept gilt als einfacher zu verstehen als das Konzept des RDF. Es fehlt hierfür jedoch eine prominente große Unterstützung, wie die des W3C.

5 Ausblick

Fakt ist, dass das Internet weiter wachsen und das Datenangebot enorm steigen wird. Man muss der eben aufgeführten Kritik entgegen halten, dass es ungedingt nötig scheint, Technologien einsetzen zu können, die sich in dieser riesigen Flut von Informationen zurecht finden und die diese voll ausschöpfen können. Das Semantic Web ist eine davon. Man kann davon ausgehen, dass eine Verbreitung zunächst bei akademischen oder wirtschaftlichen Gruppierungen stattfinden wird, die die semantische Schicht im Web für ihre Zwecke nutzen. Eine flächendeckende Ausbreitung ist aus momentaner Sicht fraglich.

Man sollte aber bedenken, dass es sich mit dem Internet am Anfang der Neunziger Jahre ähnlich verhielt. Ein Internet für den einfachen Menschen, nutzbar für alltägliche Dinge, schien zu diesem Zeitpunkt absolut undenkbar und geradezu absurd. Das wohl Wichtigste war, dass Tim Berners-Lee seine Ideen zum damaligen Zeitpunkt nicht patentieren ließ, sondern frei zur Verfügung stellte.

Aktuell fördert das deutsche Bundesministerium für Bildung und Forschung (BMBF) Software für das 1998 von WWW-Erfinder Tim Berners-Lee vorgeschlagene semantische Web in den nächsten drei Jahren mit 13,7 Millionen Euro.⁷ Aber auch in Zusammenarbeit mit den Nachbarländern wird EU-weit kräftig gefördert und geforscht.

⁷ Vgl. C't Ausgabe 06/2002

Literaturverzeichnis

W3C: <http://www.w3.org>, (Stand 12/2004).

W3C: „RDF activity page“: <http://www.w3.org/RDF/>, (Stand 12/2004).

W3C: „RDF Primer: W3C Recommendation, February 10th 2004“: <http://www.w3.org/TR/rdf-primer>, (Stand 12/2004).

W3C: „RDF Vocabulary Description Language 1.0: RDF-Schema“: <http://www.w3.org/TR/rdf-schema/>, (Stand 12/2004).

W3C: „Describing and retrieving photos using RDF and HTTP“, Note, 28 September 2000: <http://www.w3.org/TR/2000/NOTE-photo-rdf-20000928/#goals>, (Stand 01/2004).

Berners-Lee, Tim: „Semantic Web: Ontologien, Taxonomien, DAML+OIL, RDF, XML.“ Report, Trends: Semantic Web.

Wikipedia: http://de.wikipedia.org/wiki/Resource_Description_Framework, (Stand 12/2004).

Ziegler, Cai (2002): „Deus ex Machina: Das Web soll lernen, sich und uns zu verstehen.“ In: C't Magazin für Computer und Technik, Ausgabe 06/2002, Seite 132.

Haase, Broekstra, Eberhart, Volz (2004): „A comparison of RDF query languages“. In: Proceedings of the Third International Semantic Web Conference, Hiroshima, Japan.

Haase, Broekstra, Eberhart, Volz (2002): „Sesame: A Generic Architecture for Storing and Querying RDF and RDF-Schema“. In: Proceedings of the International Semantic Web Conference 2002, Sardinia, Italy.

Ünal, Akif (2002): „RDF & RDFS“, Seminar-Ausarbeitung, Universität München: <http://www11.informatik.tu-muenchen.de/lehre/seminare/seminarSW-SS2002.html>, (Stand 12/2004).

Miller, Stefan (2003): „Ziele und Visionen des Semantic Web“, Abt. Künstliche Intelligenz, Hauptseminar Techniken des Semantic Web, Wintersemester 2002/2003: <http://stefan.miller@informatik.uni-ulm.de>, (Stand 12/2004), Universität Ulm.

Beiler, Stefanie (2003): „Semantic Web“, Folien zu Seminar „Internet Dienste“ SS 03,: <http://www.informatik.uni-ulm.de/ki/Edu/Seminare/Semantic.Web/WS0203/>, (Stand 12/2004). Universität Ulm.

Universität Essen: <http://nestroy.wi-inf.uni-essen.de:8080/Lv/ibis2/folien/04.html>, (Stand 12/2004).

„Semantic Web Infotag - 18.02.2003“: <http://swit.xml-clearinghouse.de/>, (Stand 12/2004).