

Chancen und Techniken von Linked Data

STUDIENARBEIT

**von
Angelo Veltens**

15.01.2010



Chancen und Techniken von Linked Data von Angelo Veltens steht unter einer Creative Commons Namensnennung-Keine kommerzielle Nutzung-Weitergabe unter gleichen Bedingungen 3.0 Deutschland Lizenz. (<http://creativecommons.org/licenses/by-nc-sa/3.0/de/>)

Sie dürfen:

- das Werk bzw. den Inhalt vervielfältigen, verbreiten und öffentlich zugänglich machen
- Abwandlungen und Bearbeitungen des Werkes bzw. Inhaltes anfertigen

Zu den folgenden Bedingungen:

- Namensnennung — Sie müssen den Namen des Autors/Rechteinhabers in der von ihm festgelegten Weise nennen.
- Keine kommerzielle Nutzung — Dieses Werk bzw. dieser Inhalt darf nicht für kommerzielle Zwecke verwendet werden.
- Weitergabe unter gleichen Bedingungen — Wenn Sie das lizenzierte Werk bzw. den lizenzierten Inhalt bearbeiten oder in anderer Weise erkennbar als Grundlage für eigenes Schaffen verwenden, dürfen Sie die daraufhin neu entstandenen Werke bzw. Inhalte nur unter Verwendung von Lizenzbedingungen weitergeben, die mit denen dieses Lizenzvertrages identisch oder vergleichbar sind.

Nähere Informationen zur Lizenz: <http://creativecommons.org/licenses/by-nc-sa/3.0/de/>

Kontakt zum Autor: angelo.veltens@online.de

Inhaltsverzeichnis

| | | |
|----------|--|-----------|
| 1 | Einleitung | 4 |
| 2 | Der thematische Kontext von Linked Data | 5 |
| 2.1 | Vom „Web aus Dokumenten“ zum „Web aus Daten“ | 5 |
| 2.2 | Das Semantische Web | 6 |
| 3 | Linked Data aus technischer Sicht | 8 |
| 3.1 | Grundprinzipien | 8 |
| 3.1.1 | Benennung von Dingen mittels URIs | 8 |
| 3.1.2 | Dereferenzierung der URI mittels HTTP | 9 |
| 3.1.3 | Strukturierte Information anbieten | 10 |
| 3.1.4 | Andere Ressourcen verlinken | 10 |
| 3.2 | Daten erstellen | 11 |
| 3.2.1 | Das Resource Description Framework | 11 |
| 3.2.2 | Ontologien | 17 |
| 3.3 | Daten publizieren | 23 |
| 3.3.1 | Arten von Ressourcen | 23 |
| 3.3.2 | Inhaltsvereinbarung | 23 |
| 3.3.3 | URI Namensstrategien | 24 |
| 3.3.4 | Abgrenzung von Daten und Metadaten | 25 |
| 3.4 | Daten konsumieren | 26 |
| 3.4.1 | Linked Data Browser | 26 |
| 3.4.2 | SPARQL | 27 |
| 3.4.3 | Semantische Suchmaschinen | 28 |
| 3.4.4 | Data MashUps | 28 |
| 4 | Linked Data in der Praxis | 30 |
| 4.1 | DBpedia | 30 |
| 4.2 | Faviki | 31 |
| 4.3 | revyu.com | 31 |
| 4.4 | Triplify | 32 |

| | |
|--|-----------|
| 5 Chancen von Linked Data | 33 |
| 5.1 Linked Data aus wirtschaftlicher Sicht | 33 |
| 5.1.1 Linked Data als Chance für Verbraucher | 33 |
| 5.1.2 Linked Data als Chance für Unternehmen | 34 |
| 5.2 Wissenschaft und Forschung | 35 |
| 5.3 Demokratie und Staatswesen | 35 |
| 5.4 Gefahren und Risiken von Linked Data | 36 |
| | |
| 6 Zukunftsaussichten | 38 |
| 6.1 Problem „Database Hugging“ | 38 |
| 6.2 Social Semantic Web | 38 |
| 6.3 Anwendungen und Tools | 39 |
| 6.4 Vision | 40 |
| | |
| 7 Zusammenfassung und Fazit | 41 |

1 Einleitung

In dieser Studienarbeit wird das Konzept „Linked Data“ beschrieben. Dabei handelt es sich um ein Thema im Umfeld des semantischen Webs, einer erweiterten Form des World Wide Web (WWW), mit der Informationen und deren Bedeutung für Maschinen verwertbar gemacht werden sollen. Kernidee von „Linked Data“ ist es, strukturierte Daten über URIs zugänglich zu machen und untereinander zu verlinken, um so ein Web aus Daten aufzubauen und verwandte Informationen auffindbar zu machen. Die Studienarbeit ordnet „Linked Data“ in das thematische Umfeld des semantischen Webs ein und erörtert den Stand der Technik. Anwendungsgebiete und Chancen werden aufgezeigt. Die Studienarbeit soll darüber hinaus praktische Anwendungsmöglichkeiten für „Linked Data“ darlegen und die Zukunftsaussichten bewerten.

In Kapitel 2 wird zunächst der Begriff „Linked Data“ eingeführt und der Wandel von einem „Web aus Dokumenten“ zu einem „Web aus Daten“ skizziert. Ebenfalls wird die Thematik des semantischen Webs kurz eingeführt und Linked Data in dieses Umfeld eingeordnet.

Kapitel 3 befasst sich anschließend mit den technischen Grundlagen von Linked Data. Die von Tim Berners-Lee eingeführten Grundprinzipien werden dargelegt und es wird aufgezeigt, wie Linked Data erstellt, publiziert und konsumiert werden kann.

Mit bereits bestehenden praktischen Einsatzmöglichkeiten von Linked Data befasst sich Kapitel 4. Anhand einiger existierender Projekte, die Linked Data bereitstellen oder verarbeiten, wird der Stand der praktischen Nutzung aufgezeigt.

In Kapitel 5 diskutiert diese Studienarbeit die Chancen die ein Web aus strukturierten und verknüpften Daten bietet. Dabei wird sowohl auf die wirtschaftlichen Chancen für Verbraucher und Unternehmen eingegangen, als auch die Vorteile für Wissenschaft und Forschung. Zudem wird dargelegt, wie Linked Data ein transparentes und demokratisches Staatswesen fördern kann. Darüber hinaus wird auch auf mögliche Gefahren von Linked Data eingegangen.

Kapitel 6 befasst sich mit den Zukunftsaussichten von Linked Data. Es betrachtet dabei zum einen die nächsten Schritte die für die Fortentwicklung des „Web aus Daten“ notwendig sind und geht zum anderen auf die Vision ein, die hinter der Idee von Linked Data steht.

Abschließend fasst Kapitel 7 diese Studienarbeit kurz zusammen und gibt ein Fazit ab.

2 Der thematische Kontext von Linked Data

„Linked Data“ bedeutet „verknüpfte Daten“. Tim Berners-Lee hat diesen Begriff 2006 in seinem Dokument „Design Issues: Linked Data“ eingeführt. Linked Data kann als Teilgebiet des Semantischen Webs betrachtet werden. Dabei handelt es sich um eine Erweiterung des WWW, welche das Ziel hat, Informationen im Web maschinenlesbar und automatisiert verarbeitbar zu machen und miteinander zu verknüpfen. Berners-Lee betont die Bedeutung von Links im Semantischen Web:

The Semantic Web isn't just about putting data on the web. It is about making links, so that a person or machine can explore the web of data. With linked data, when you have some of it, you can find other, related, data.[1]

Das WWW, wie wir es heute kennen, ist ein Web aus *Dokumenten*. Linked Data hat das Ziel, ein Web aus *Daten* zu schaffen. Dieses Kapitel grenzt zunächst die Idee eines solchen Datenwebs vom bestehenden WWW ab und ordnet sie dann in die Thematik des Semantischen Webs ein.

2.1 Vom „Web aus Dokumenten“ zum „Web aus Daten“

Die ursprüngliche Intention des WWW war, Forschungsergebnisse leichter zugänglich zu machen. Auf wissenschaftliche Dokumente sollte ortsunabhängig zugegriffen werden können. Die Dokumente werden dazu mittels Hypertext beschrieben, über eine eindeutige Adresse im Internet abrufbar gemacht und miteinander verlinkt. Über das Hypertext Transfer Protocol (HTTP) können Browser die Dokumente abrufen und die durch Hypertext Markup Language (HTML) beschriebene Dokumentenstruktur grafisch darstellen. Bis heute hat sich an diesen Grundlagen des WWW wenig geändert. Selbst moderne Webseiten mit multimedialen Inhalten, dynamischem Nachladen mittels Ajax und Interaktionmöglichkeiten für die Nutzer sind HTML-Dokumente.

Dieses Web ist auf Menschen zugeschnitten. Gutes Webdesign vorausgesetzt, sind die Dokumente leicht für Menschen lesbar und über Verlinkungen lassen sich verwandte Seiten abrufen. Sobald es jedoch um eine maschinelle Auswertung der Inhalte im Web geht, scheitert der aktuelle Ansatz. Zwar wird das WWW von Webcrawlern durchsucht

und indiziert, doch wird dabei nicht die Bedeutung der Inhalte erfasst. Suchmaschinen finden deshalb heute nur Inhalte im Web, aber keine Antworten. Wer eine Frage in eine Suchmaschine eingibt, der findet Webseiten die diese Frage enthalten. Die Antwort wird dabei bestenfalls vom Nutzer entdeckt, weil sie zufällig im gleichen Dokument steht oder nur wenige Links entfernt. Die Suchmaschine hat diese Antwort weder gesucht noch gefunden.

Das WWW ist ein Web aus Dokumenten. Diese Dokumente können nützliche Daten enthalten. Die Bedeutung der Daten kann jedoch nur der menschliche Nutzer erkennen, der eine Webseite liest und die dargestellten Informationen interpretiert. Für ein Computerprogramm ist nicht ohne weitere ersichtlich, ob eine HTML-Tabelle Daten über den Jahresniederschlag oder Einwohnerzahlen enthält. Selbst wenn die Tabelle entsprechend beschriftet ist, kann ein Programm dadurch noch nicht die Bedeutung erkennen. Erschwerend hinzu kommt die Vielfalt an möglichen Darstellungsformen. Auf absehbare Zeit ist es unwahrscheinlich, dass Software annähernd das Textverständnis eines Menschen vorweisen kann. Mensch und Maschine sprechen unterschiedliche Sprachen.

Im Informationszeitalter entsteht das Dilemma, dass die Fülle an Informationen immer größer wird, gleichzeitig aber nur Menschen diese Informationen sinnvoll auswerten können. Um die Informationsmassen auf Dauer verarbeiten zu können, müssen strukturierte Daten im Web verfügbar gemacht und auf der Ebene ihrer Bedeutung miteinander in Beziehung gesetzt werden. Dabei geht es nicht mehr bloß um Dokumente, sondern um Daten über alle möglichen Dinge, wie zum Beispiel Städte, Personen, Ereignisse, Medikamente, Organisationen und vieles mehr. Aus verknüpften Daten können sich neue, vorher unbekannte Informationszusammenhänge ergeben die zu neuen Erkenntnissen führen. Ein solches Web aus Daten käme somit der Wissenschaft und Forschung ebenso zugute, wie der Wirtschaft und der Gesellschaft im Ganzen.

2.2 Das Semantische Web

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. [2]

Linked Data bedeutet, strukturierte Daten im Web verfügbar zu machen und miteinander zu verknüpfen. Damit stellt Linked Data einen wichtigen Bestandteil des Semantischen Webs dar, ist jedoch nicht mit diesem gleichzusetzen. Während beim Semantischen Web auf einem hohen Level auch die Repräsentation von Logik und Wissen thematisiert wird, geht es bei Linked Data um die Veröffentlichung von verknüpften Daten mit niedrigen Einstiegshürden:

In contrast to the full-fledged Semantic Web vision, linked data is mainly about publishing structured data in RDF using URIs rather than focusing on the ontology level or inference. This simplification - just as the Web simplified the established academic approaches of Hypertext systems - lowers the entry barrier for data provider, hence fosters a wide spread adoption. [3]

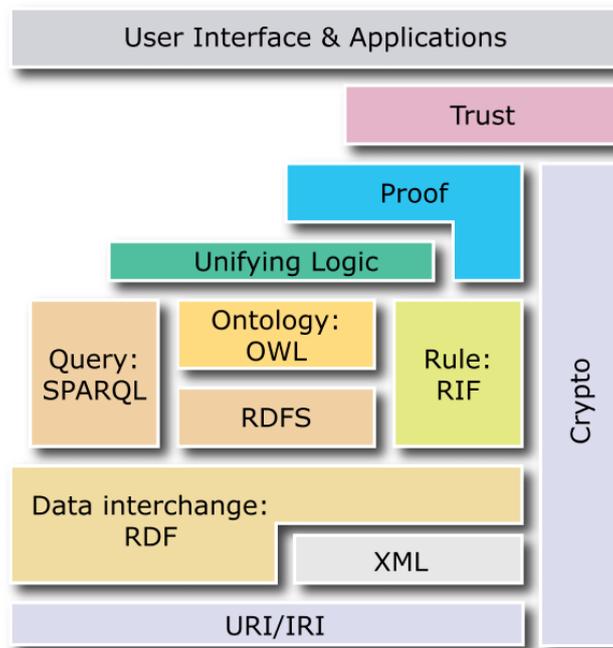


Abbildung 2.1: Schichtenmodell des Semantischen Webs. Quelle: World Wide Web Consortium (<http://www.w3.org/2007/03/layerCake.png>)

Abbildung 2.1 zeigt ein vom World Wide Web Consortium (W3C) definiertes Schichtenmodell des Semantischen Webs. Linked Data betreffende Technologien sind in den unteren Schichten anzutreffen. Dazu zählen insbesondere RDF, eine Sprache zur formalen Beschreibung von Ressourcen, und URI, zur Identifizierung von Ressourcen. Zumindest Berührungspunkte mit Linked Data haben als Abfragesprache SPARQL und als Ontologie-Beschreibungssprachen RDFS und OWL. Diese Technologien werden in der vorliegenden Studienarbeit behandelt.

Darüber hinaus zählt zum Semantischen Web auch die wissenschaftliche Definition von Ontologien zur Beschreibung der Wirklichkeit und der formalen Repräsentation von Wissen. Dies ermöglicht das Ziehen von logischen Schlüssen aus dem so definierten Wissen und die Überprüfung von Behauptungen nach formalen Regeln. Das Semantische Web beschäftigt sich auch mit der Problemstellung des Vertrauens: Der Frage, welche Informationen korrekt sind und welche Quellen vertrauenswürdig. Diese weitergehenden Themen bilden den Kontext in dem Linked Data in dieser Studienarbeit betrachtet wird.

3 Linked Data aus technischer Sicht

In diesem Kapitel werden zunächst die von Berners-Lee formulierten Grundprinzipien von Linked Data vorgestellt. Anschließend wird erörtert, wie Linked Data erstellt, publiziert und konsumiert werden kann.

3.1 Grundprinzipien

Berners-Lee nennt vier Regeln um Daten im Web verfügbar zu machen. Gleichzeitig weist er darauf hin, dass diese Regeln vielmehr ein erwartetes Verhalten definieren, als einen zwingenden Rahmen vorgeben. Niemand muss diese Regeln einhalten - doch ist jede Missachtung dieser Regeln laut Berners-Lee eine verpasste Chance, Daten miteinander zu verbinden. Die vier Regeln lauten:

1. Use URIs as names for things
2. Use HTTP URIs so that people can look up those names.
3. When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL)
4. Include links to other URIs. so that they can discover more things. [1]

Die Regeln sind grundsätzlich sehr einfach gehalten, werfen bei der Umsetzung in die Praxis jedoch einige Fragen und Problemstellungen auf. In den folgenden Abschnitten werden die Regeln zunächst näher erläutert und im weiteren Verlauf des Kapitels die technischen Möglichkeiten der Umsetzung besprochen.

3.1.1 Benennung von Dingen mittels URIs

Die erste Regel besagt, dass URIs verwendet werden sollen, um Dinge zu benennen. URI steht für Uniform Resource Identifier. Es handelt sich dabei um eine nach bestimmten Vorgaben formatierte Zeichenfolge, die als eindeutiger Bezeichner für Ressourcen im Internet dient. Das Format einer URI ist in RFC 3986 [4] definiert; der grundlegende Aufbau lautet wie folgt:

```
URI = scheme ":" hier-part [ "?" query ] [ "#" fragment ]
```

Da URI ein erweiterbares Namenssystem darstellt, wird über das Schema (**scheme**) definiert, welchen zusätzlichen Richtlinien der Aufbau der URI zu folgen hat. Oft vorfindbare Schemas sind zum Beispiel **http**, **ftp**, **file** und **mailto**.

Der **hier-part** besteht aus Anbieter (**authority**) und Pfad (**path**), wobei nur eines von beidem zwingend in einer URI angegeben werden muss. Der **authority**-Teil gibt an, wer den durch die URI definierten Namen verwaltet. Der Pfad identifiziert ggf. zusammen mit dem optionalen Abfrageteil (**query**) die tatsächliche Ressource. Der ebenfalls optionale Abschnitt **fragment** kann genutzt werden, um einen speziellen Teil einer Ressource zu referenzieren. Abbildung 3.1 stellt einige beispielhafte URIs dar und verdeutlicht durch farbliche Hervorhebung deren Aufbau.

```
http://foo.example.org:8080/path/to/resource?id=123#chp1
urn:example:foo:bar:item
ftp://192.168.0.7/foobar.txt
file://localhost/tmp
scheme authority path query fragment
```

Abbildung 3.1: Einige Beispiels-URIs und deren Aufbau

Berners-Lee spricht in seiner Abhandlung jedoch nicht von Ressourcen, die durch URIs identifiziert werden sollen, sondern ganz allgemein von *Dingen*. Diesem Gedanken liegt die Idee zugrunde, aus dem „Web aus Dokumenten“ ein „Web aus Daten“ zu machen. Zwar liegen die Daten in diesem Web auch in Form von Dokumenten vor. Die Dinge die sie beschreiben müssen jedoch ebenso eindeutig referenzierbar sein, wie die Dokumente selbst. Deshalb braucht jedes „Ding“ (d.h. jedes Buch, jedes Gebäude, jede Person, jedes chemische Element . . .), über das Daten im Web abgelegt werden, eine eindeutige Kennzeichnung um ihm diese Daten zuzuordnen. Wenn im Folgenden von Ressourcen gesprochen wird, schließt das all diese Dinge ein.

3.1.2 Dereferenzierung der URI mittels HTTP

Berners-Lee fordert weiter, dass HTTP-URIs verwendet werden, damit der durch die URI bestimmte Name nachgeschlagen werden kann. HTTP-URIs haben sich im WWW bewährt um Dokumente eindeutig zu identifizieren und abzurufen.

Der **authority**-Teil muss bei einer HTTP-URI vorhanden sein. Er enthält eine Hostadresse in Form eines Fully Qualified Domain Name (FQDN). Der hierarchische Aufbau des FQDN macht die Auflösung über das Domain Name System (DNS) möglich, welche dem anfragenden Client die IP-Adresse des Hosts liefert. Der Client kann nun eine HTTP-Verbindung mit dem Host aufbauen um die durch die URI identifizierten Daten abzufragen.

HTTP und DNS sind Standardtechnologien, die bereits erfolgreich im WWW angewandt werden. Für eine nähere Betrachtung sei auf „Computer Networking: A Top-Down Approach“ [5] verwiesen. Es ist naheliegend diese Technologien ebenfalls für Linked Data einzusetzen, da sie bereits voll implementiert und ohne Anpassung für diesen Zweck nutzbar sind.

Jedoch ist dringend davon abzuraten, andere Formen von URIs zu verwenden. Zur Identifikation von Büchern scheint zum Beispiel zunächst eine URI der Form `urn:isbn:3861990016` sinnvoll, da sich Bücher über die ISBN eindeutig identifizieren lassen. Diese URI lässt sich allerdings nicht auflösen, das heißt es gibt keine Information über einen Server, der Informationen über das Buch enthält. Die URI ist somit aus Sicht von Linked Data unbrauchbar.

3.1.3 Strukturierte Information anbieten

Die dritte Regeln besagt, dass über eine solche URI strukturierte Informationen abrufbar sein sollen. Berners-Lee spricht an dieser Stelle noch etwas unpräzise von „nützlichen“ Informationen, verweist jedoch bereits auf Standards wie RDF und SPARQL. Auf beide Technologien wird dieses Kapitel noch eingehen. Zunächst reicht es zu wissen, dass die Informationen strukturiert und leicht weiterverarbeitbar zur Verfügung gestellt werden müssen, um nützlich zu sein. Diese Forderung schließt zum Beispiel aus, unter der URI Daten als ZIP-Archiv auszuliefern, oder Rohdaten, ohne den Kontext ihrer Bedeutung.

3.1.4 Andere Ressourcen verlinken

Was für die Dokumente im WWW seit jeher selbstverständlich ist, wird in der letzten Regel auch für Daten gefordert: Es reicht nicht aus, Daten ins Netz zu stellen, sondern die Daten müssen auch verlinkt werden. Was das technisch bedeutet, wird in Abschnitt 3.2.1 erläutert. Es ist jedoch wichtig zu erkennen, dass erst die Verknüpfung von Daten, diese mit zusätzlichem Wert anreichert. Dies sei an einem kurzen Beispiel verdeutlicht: Man könnte zu der Stadt Karlsruhe verschiedene Informationen ins Internet stellen, wie zum Beispiel die Einwohnerzahl, die Fläche und das Gründungsjahr. Weiterhin könnte man in diesem Datenbestand auch mitteilen, dass Karlsruhe am Rhein liegt. Weitere Informationen bleiben dem Nutzer des Datensatzes jedoch verborgen. Erst wenn statt der bloßen Nennung des Rheins, auf eine Ressource im Web verlinkt wird, die den Rhein beschreibt, wird ein Mehrwert generiert. Durch Folgen des Links können nähere Informationen über den Rhein gefunden werden, wie zum Beispiel die Information, dass es sich dabei um einen Fluss handelt, seine Länge und weitere Städte die ebenfalls am Rhein liegen. Durch die Verknüpfung der Daten entsteht ein Web, das nicht nur die Informationen enthält, die unmittelbar publiziert wurden, sondern auch neue, vorher unbekannte Zusammenhänge aufdecken kann.

3.2 Daten erstellen

Nachdem nun die Grundprinzipien von Linked Data bekannt sind, wird in diesem Abschnitt beschrieben, wie Linked Data erstellt werden kann. Dazu wird das Resource Description Framework und der Begriff Ontologie eingeführt.

3.2.1 Das Resource Description Framework

Dokumente die im WWW veröffentlicht werden, sind durch die Auszeichnungssprache HTML formal beschrieben. HTML gliedert ein Dokument in Überschriften, Absätze, Bilder und mehr. Bestimmte Teile eines HTML-Dokuments können zudem als Links ausgezeichnet werden, um auf andere Inhalte im Web zu verweisen. Die formale Beschreibung ermöglicht es dem Browser, das Dokument entsprechend der Gliederung so aufzubereiten und anzuzeigen, dass es vom menschlichen Nutzer leicht gelesen werden kann. Ebenso müssen die Daten, die im Web veröffentlicht werden sollen, formal beschrieben werden. Der Fokus liegt hier jedoch darauf, die Daten so zu beschreiben, dass sie leicht von Maschinen gelesen und verarbeitet werden können.

In der Geschichte der Informatik sind zahlreiche Formate für den Datenaustausch und die automatisierte Weiterverarbeitung dieser Daten entstanden. Ein einfaches Format ist zum Beispiel CSV. Die Daten liegen hierbei in einer Textdatei und sind durch ein Trennzeichen (zum Beispiel Komma oder Semikolon) voneinander getrennt. Dabei ist jedoch völlig unbekannt, was diese Daten beschreiben. Dieser Informationsmangel betrifft sowohl einen Datensatz im Gesamten („Was beschreibt dieser Datensatz?“), als auch einzelne Spalten in der Datei („Was bedeutet der Wert in Spalte 3?“). Tabelle 3.1 verdeutlicht dieses Problem.

Tabelle 3.1: Daten einer Beispiel-CSV-Datei. Aus der Datei geht die Bedeutung der Daten nicht hervor.

| | | | |
|--------|-------|----|---------|
| 891.82 | 030 | B | 3431700 |
| 173.46 | 0721 | KA | 288917 |
| 105.02 | 0261 | KO | 105888 |
| 97.75 | 06131 | MZ | 200234 |

Die Bedeutung der Daten muss zwischen den Kommunikationspartnern abgestimmt werden und in die datenverarbeitenden Systeme implementiert werden. Es kann also passieren, dass zwei unterschiedliche Systeme den gleichen Datensatz völlig unterschiedlich interpretieren. Selbst eine neuere Version des gleichen Systems könnte die Daten anders verstehen.

Eine brauchbare formale Beschreibung für Linked Data muss folgende Fragen beantworten:

1. Was beschreiben die Daten?
2. Welche Bedeutung haben die einzelnen Datenwerte?
3. Wie lauten die Datenwerte?

Die formale Beschreibung von Ressourcen mittels RDF gewährleistet genau dies.

Grundkonzept von RDF

RDF steht für Resource Description Framework und ist eine Sprache, mit deren Hilfe Informationen über Ressourcen formal beschrieben werden können. Die Informationen werden bei RDF in Form von sogenannten Tripeln notiert. Ein Tripel setzt sich zusammen aus Subjekt, Prädikat und Objekt. Subjekt und Objekt werden über das Prädikat in Beziehung gesetzt. Das Subjekt ist dabei die Ressource welche beschrieben wird. Das Prädikat gibt an, auf welchen Sachverhalt sich die Information bezieht, das heißt, welche Bedeutung das Objekt in Bezug auf das Subjekt hat. Das Objekt ist entweder ein einfacher Datenwert oder der Verweis auf eine weitere Ressource. Tabelle 3.2 enthält einige an natürliche Sprache angelegte RDF-Tripel.

Tabelle 3.2: Beispiele für RDF-Tripel, angelehnt an natürliche Sprache.

| Subjekt | Prädikat | Objekt |
|-----------|-------------------|--------|
| Karlsruhe | hat Einwohnerzahl | 288917 |
| Karlsruhe | liegt am | Rhein |
| Koblenz | hat Einwohnerzahl | 105888 |
| Koblenz | liegt am | Rhein |
| Koblenz | liegt am | Mosel |
| Rhein | hat Länge | 1324 |
| Mosel | hat Länge | 544 |
| Mosel | mündet in | Rhein |

Diese Informationen lassen sich auch als gerichteter Graph darstellen, einem sogenannten RDF-Graphen. In Abbildung 3.2 werden Ressourcen als Ellipse und Datenwerte als Rechtecke dargestellt. Die Prädikate finden sich in Form von Pfeilen wieder, an deren Beginn sich jeweils das Subjekt, und an dessen Ende das Objekt befindet. Durch traversieren des Graphen lassen sich neue Informationen und Zusammenhänge finden.

RDF gibt nur ein Modell, jedoch keine Syntax für die Notation der Tripel vor. RDF kann zum Beispiel als XML oder in der von Tim Berners-Lee entwickelten Kurzschreibweise Notation 3 dargestellt werden. Weiterhin ist es mit RDFa möglich, RDF in XHTML einzubetten. Diese drei Notationsweisen werden in den nächsten Abschnitten dargelegt. Der Schwerpunkt liegt dabei darauf, die unterschiedlichen Notationen kurz vorzustellen.

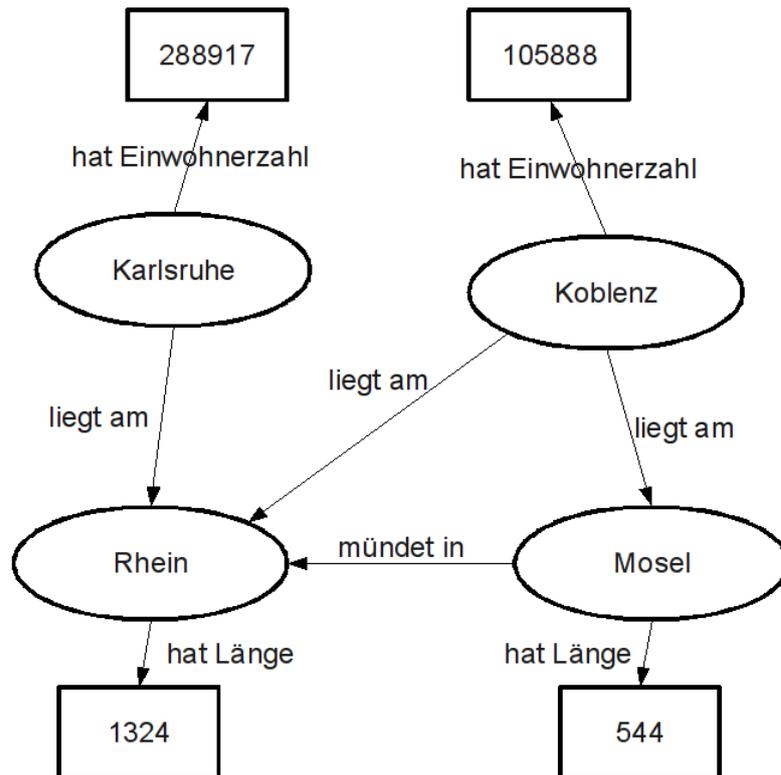


Abbildung 3.2: Darstellung der Daten aus Tabelle 3.2 in einem RDF-Graphen

Auf weiterführende, detailliertere Informationen wird innerhalb des jeweiligen Abschnitts verwiesen.

RDF/XML

RDF wird oft mit RDF/XML verwechselt, das RDF häufig in seiner XML-Notation dargestellt wird. RDF ist aber wie im vorherigen Abschnitt beschrieben nur ein abstraktes Modell, das sich in vielerlei Weisen darstellen lässt. Die Grundstruktur eines RDF/XML-Dokuments ist aus Listing 3.1 ersichtlich.

Listing 3.1: Grundstruktur eines RDF/XML-Dokuments

```

1 <rdf:RDF
2   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3
4   <rdf:description rdf:about="http://example.org/article/0815">
5
6   </rdf:description>
7
8 </rdf:RDF>

```

In den Zeilen 1 und 2 steht das Wurzelement `<rdf:RDF>` inklusive der XML-typischen Namensraum-Deklarationen. In der vierten Zeile wird mit `<rdf:description>` die

Beschreibung einer Ressource eingeleitet. Das Attribut `rdf:about` benennt die Ressource um die es geht, in diesem Fall also `http://example.org/article/0815`. Die Tags werden in den Zeilen 6 und 8 entsprechend wieder geschlossen. In Listing 3.2 werden Daten zu diesem Artikel hinterlegt:

Listing 3.2: Beschreibung eines Artikels in RDF/XML

```
1 <rdf:RDF
2   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3   xmlns:dc="http://purl.org/dc/terms/"
4
5 <rdf:description rdf:about="http://example.org/article/0815">
6   <rdf:type rdf:resource=
7     "http://purl.org/dc/terms/BibliographicResource"/>
8   <dc:title>Metasyntaktische Variablen</dc:title>
9   <dc:creator>Max Muster</dc:creator>
10  <dc:modified>2009-12-07</dc:modified>
11 </rdf:description>
12
13 </rdf:RDF>
```

Es fällt zunächst auf, dass in Zeile 3 ein weiterer Namensraum definiert wird. Hinter `http://purl.org/dc/terms/` verbirgt sich eine Ontologie, die den Dublin Core Standard formal beschreibt. Nähere Informationen zu Ontologien bietet Abschnitt 3.2.2. Dieser Namensraum stellt eine Reihe neuer Tags zur Verfügung, mit denen Informationen zu der Ressource ausgedrückt werden können. Einige dieser Tags werden in den Zeilen 6 bis 8 verwendet. `<rdf:type>` bestimmt den Typ der beschriebenen Ressource. In dem Beispiel wird die Ressource als eine `BibliographicResource` (Bibliographische Ressource) gekennzeichnet. Unter der URI `http://purl.org/dc/terms/BibliographicResource` lassen sich genauere Informationen (ebenfalls in Form von RDF) abrufen, was darunter zu verstehen ist: „A book, article, or other documentary resource“. Durch das Tag `<dc:title>` wird der Titel der bibliographischen Ressource bestimmt, `<dc:creator>` legt fest, dass der Verfasser der Ressource Max Muster heißt. Schlussendlich bestimmt `<dc:modified>` das Datum, wann die Ressource zuletzt geändert wurde.

Alternativ lässt sich eine Aussage über den Typ auch implizit durch die Verwendung eines entsprechenden Eltern-Tags treffen. Listing 3.3 nutzt `<dc:BibliographicResource>` anstelle von `<rdf:description>` um die Ressource zu beschreiben. Dadurch erhält sie implizit den Typ `http://purl.org/dc/terms/BibliographicResource`.

Listing 3.3: Implizite Bestimmung des Typs durch Wahl des Eltern-Tags

```
1 <rdf:RDF
2   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3   xmlns:dc="http://purl.org/dc/terms/"
4   <dc:BibliographicResource rdf:about="http://example.org/article/0815">
```

```

3 xmlns:dc="http://purl.org/dc/terms/">
4
5 <dc:BibliographicResource
6   rdf:about="http://example.org/article/0815">
7   <dc:title>Metasyntaktische Variablen</dc:title>
8   <dc:creator>Max Muster</dc:creator>
9   <dc:modified>2009-12-07</dc:modified>
10 </dc:BibliographicResource>
11
12 </rdf:RDF>

```

Bei dem bisherigen Beispiel kann aber noch keine Rede von Linked Data sein, da Links zu anderen Ressourcen fehlen. Ein Kandidat für die Verlinkung ist in diesem Falle der Verfasser. Allein ein Name ist wenig aussagekräftig, Listing 3.4 ersetzt ihn deshalb durch eine URI, die die Person Max Muster identifiziert. Dabei ist zu beachten, dass die URI nicht als Tag-Inhalt verwendet wird, sondern über das Attribut `rdf:resource` angegeben werden muss.

Listing 3.4: Verlinkung auf eine andere Ressource

```

1 <rdf:RDF
2   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3   xmlns:dc="http://purl.org/dc/terms/">
4
5   <dc:BibliographicResource
6     rdf:about="http://example.org/article/0815">
7     <dc:title>Metasyntaktische Variablen</dc:title>
8     <dc:creator rdf:resource="http://example.org/creator/maxmuster" /
9       >
10    <dc:modified>2009-12-07</dc:modified>
11 </dc:BibliographicResource>
12 </rdf:RDF>

```

Die hier aufgeführten Beispiele geben einen Überblick über die Grundfunktionalität von RDF/XML, die für einen Einstieg in das Thema Linked Data ausreichend sind. Für einen tieferen Einblick in die XML-Notation von RDF sei auf das Tutorial von w3schools.com verwiesen. [6]

Notation 3

Tim Berners-Lee hat für RDF eine Kurzschreibweise eingeführt die für Menschen leichter zu lesen und zu schreiben ist, als die XML-Notation. Diese Notation 3 (N3) getaufte Schreibweise reiht grundsätzlich nur Subjekt, Prädikat und Objekt eines RDF-Tripels

aneinander. In Listing 3.5 ist der gleiche Sachverhalt wie in Listing 3.4 niedergeschrieben, jedoch in N3-Notation.

Listing 3.5: RDF-Tripel in Notation 3

```
1 @prefix dc: <http://purl.org/dc/terms/>.
2
3 <http://example.org/article/0815>
4   dc:title "Metasyntaktische_Variablen";
5   dc:creator <http://example.org/creator/maxmuster>;
6   dc:modified "2009-12-07";
7   a dc:BibliographicResource.
```

In Zeile 1 wird das Prefix `dc` bestimmt, was der XML-Namensraumdeklaration gleichkommt. In Zeile 3 wird das Subjekt benannt und in den darauf folgenden Zeilen beschrieben. Die Prädikat-Objekt-Paare beziehen sich solange auf dieses Subjekt, bis die Beschreibung der Ressource durch einen Punkt „.“ abgeschlossen wird. Paare die sich auf das gleiche Subjekt beziehen werden durch ein Semikolon unterbrochen. Datenwerte, wie zum Beispiel der Titel in Zeile 4, werden in Anführungszeichen gesetzt. Links zu anderen Ressourcen erfolgen durch die Notation der URI in spitzen Klammern, wie der Link zum Verfasser in Zeile 5. Zeile 8 bestimmt den Typ der Ressource.

N3 ist trotz seiner Kürze weitaus mächtiger als im Rahmen dieser Studienarbeit beschrieben. Einen guten Einstieg in diese Notation bietet die Einführung des W3C. [7]

RDFa

RDFa erweitert XHTML um die Möglichkeit Metadaten in Form von RDF in Web-Dokumente einzubetten. Dazu wurde XHTML um einige Attribute erweitert, die in Tabelle 3.3 beschrieben sind. Die in Tabelle 3.4 gelisteten Attribute sind bereits in XHTML enthalten und werden für RDFa in der dort beschriebenen Weise genutzt. Einen detaillierteren Einblick bietet der Artikel „RDFa in XHTML“. [8]

Listing 3.6 verdeutlicht, wie die Daten der in den vorangegangenen Abschnitten beschriebene bibliografische Ressource mittels RDFa direkt in die Ressource selbst eingebettet werden können. Das Beispiel geht davon aus, dass es sich bei der Ressource um einen in HTML beschriebenen Online-Artikel handelt. Dieser wird mittels RDFa um Metadaten angereichert.

Listing 3.6: Ein mit RDFa um Metadaten angereicherter Online-Artikel

```
1 <div xmlns:dc="http://purl.org/dc/terms/"
2   about="http://example.org/article/0815"
3   typeof="dc:BibliographicResource">
4   <h1 property="dc:title">Metasyntaktische Variablen</h1>
5   <div>
```

```

6   Autor:
7   <span rel="dc:creator"
8       resource="http://example.org/creator/maxmuster">
9       Max Muster
10  </span>
11  Letzte Änderung:
12  <span property="dc:modified">2009-12-07</span>
13 </div>
14 <div>
15 Text des Artikels ...
16 </div>
17 </div>

```

Es werden gewöhnliche HTML-Elemente verwendet um den Artikel auszuzeichnen. In Zeile 1 beginnt ein `<div>`, welches den Artikel komplett umschließt. In den Attributen wird der Dublin-Core-Namensraum deklariert und über das `about`-Attribut wird bestimmt, dass der Inhalt des Tags die Ressource `http://example.org/article/0815` beschreibt. Über `typeof` wird der gleiche Typ festgelegt, wie schon in den vorangegangenen Beispielen. Die Überschrift `<h1>` in Zeile 3 sagt über `property="dc:title"` aus, dass sie den Titel dieser Ressource gemäß Dublin Core darstellt. In Zeile 7 verweist ein `` auf den Verfasser der Ressource. Hier wird, im Gegensatz zu Zeile 12, nicht der Datenwert innerhalb des Tags als Objekt verwendet, sondern die Ressource auf die über das Attribut `resource` verwiesen wird. In Zeile 7 wird daher auch das Attribut `rel` zur Bestimmung des Prädikats genutzt, in Zeile 12 jedoch `property` (Vgl. dazu Tab. 3.3 und 3.4).

Tabelle 3.3: Für RDFa neu in XHTML eingeführte Attribute und ihre Bedeutung.

| Attribut | Bedeutung |
|-----------------------|--|
| <code>about</code> | Benennt das Subjekt eines RDF-Tripels |
| <code>property</code> | Benennt das Prädikat eines RDF-Tripels, falls das Objekt ein Datenwert ist |
| <code>resource</code> | URI die das Objekt eines RDF-Tripels identifiziert |
| <code>datatype</code> | Bestimmt den Datentyp eines Datenwerts. |
| <code>typeof</code> | Bestimmt den Typ des Subjekts |

3.2.2 Ontologien

Laut einer Definition von Tom Gruber sind Ontologien in den Computerwissenschaften die Definition einer Menge von Begrifflichkeiten, mit deren Hilfe eine bestimmte Wissensdomäne modelliert werden kann [9]. In Bezug auf Linked Data stellen Ontologien

Tabelle 3.4: In XHTML enthaltene Attribute die für RDFa verwendet werden.

| Attribut | Bedeutung |
|----------|---|
| rel | Benennt das Prädikat eines RDF-Tripels, falls das Objekt eine Ressource ist |
| rev | Wie rel, jedoch wird die Beziehung in Umgekehrter Richtung ausgedrückt. |
| content | Objekt in Form eines Datenwerts |
| href | Objekt in Form einer anderen Ressource |
| src | Objekt in Form einer eingebetteten Ressource |

ein Vokabular bereit, mit deren Hilfe Beziehungen zwischen Ressourcen ausgedrückt werden können. Ontologien stellen die Prädikate für RDF-Tripel zur Verfügung. Zwei verbreitete Ontologien werden in diesem Abschnitt vorgestellt. Eine davon, Dublin Core, wurde bereits im Rahmen der Beispiele von Abschnitt 3.2.1 verwendet. Außerdem werden mit RDFS und OWL zwei Ontologie-Beschreibungssprachen kurz vorgestellt.

Dublin Core

Dublin Core ist ein Metadatenstandard, der schon unabhängig von Linked Data im Web Verbreitung gefunden hat. Ursprünglich wurde Dublin Core für die Verwendung in HTML-Metadaten entwickelt. Mit dem `<meta>`-Tag können in HTML Metadaten zu einem Web-Dokument hinterlegt werden, um zum Beispiel von Suchmaschinen ausgewertet zu werden. Das Tag verwendet dabei die Attribute `name` und `content` um die Daten als Schlüssel-Wert-Paare abzulegen. Listing 3.7 verdeutlicht dies an einem Beispiel, welches den Verfasser eines Dokuments benennt.

Listing 3.7: Hinterlegung von Metainformationen mit HTML

```
1 <meta name="creator" content="Max_Muster" />
```

Ein Problem bei diesem Verfahren entsteht dadurch, dass HTML den Wertebereich des `name`-Attributs technisch nicht einschränkt. In unterschiedlichen Web-Dokumenten sind deshalb auch oft unterschiedliche Schlüsselattribute für die Metainformation zu finden. Der Verfasser eines Dokuments könnte zum Beispiel einmal als `creator` und ein anders Mal als `author` bezeichnet werden. Weitere Divergenz entsteht durch die Verwendung unterschiedlicher Sprachen. Listing 3.8 verdeutlicht dieses Problem anhand einiger Beispiele.

Listing 3.8: Probleme mit unterschiedlichen Werten im Attribut `name`

```
1 <meta name="creator" content="Max_Muster" />
2
3 <meta name="author" content="Max_Muster" />
4
```

```
5 <meta name="autor" content="Max_Muster" />
```

Um dem entgegen zu wirken, legt Dublin Core einen Standard für den Inhalt des `name`-Attributs fest. Tabelle 3.5 beschreibt einige dieser sogenannten Felder und ihre Bedeutung.

Tabelle 3.5: Einige durch Dublin Core standardisierte Felder.

| Feld | Bedeutung |
|------------------------|--------------------------------|
| <code>title</code> | Titel des Dokuments |
| <code>creator</code> | Verfasser des Dokuments |
| <code>publisher</code> | Herausgeber des Dokuments |
| <code>created</code> | Erstellungsdatum des Dokuments |
| <code>modified</code> | Datum der letzten Änderung |

Dublin Core ist nicht auf die Verwendung innerhalb von HTML beschränkt, sondern wird zum Beispiel auch im Open-Document-Format (ODF) genutzt. Da Dublin Core mittlerweile auch formal in einer Ontologie-Beschreibungssprache verfasst wurde, können die Dublin Core Felder als Prädikate innerhalb von RDF-Tripeln genutzt werden. Ein Beispiel findet sich in Listing 3.4 auf Seite 15. Nähere Informationen zu Dublin Core finden sich auf der Website der Dublin Core Metadata Initiative [10].

FOAF

Eine weitere, verbreitete Ontologie ist FOAF. Mit FOAF können Personen beschrieben und Freundschaftsbeziehungen zwischen diesen Personen ausgedrückt werden. Anhand des Beispiels in Listing 3.9 werden im Folgenden wesentliche Elemente von FOAF vorgestellt. Für weitere Details sei auf die FOAF-Spezifikation verwiesen [11].

Listing 3.9: Beschreibung einer Person mittels FOAF

```
1 <rdf:RDF
2   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3   xmlns:foaf="http://xmlns.com/foaf/0.1/">
4   <foaf:Person rdf:about="#me">
5     <foaf:title>Herr</foaf:title>
6     <foaf:name>Max Muster</foaf:name>
7     <foaf:givenname>Max</foaf:givenname>
8     <foaf:family_name>Muster</foaf:family_name>
9     <foaf:mbox_sha1sum>f995...41</foaf:mbox_sha1sum>
10    <foaf:homepage rdf:resource="http://maxmuster.example.org"/>
11    <foaf:knows
12      rdf:resource="http://example.org/creator/franzmayer" />
```

```

13     <foaf:knows>
14         <foaf:Person>
15             <foaf:name>Miriam Muster</foaf:name>
16             <foaf:mbox_sha1sum>0ad0...30e</foaf:mbox_sha1sum>
17         </foaf:Person>
18     </foaf:knows>
19 </foaf:Person>
20 </rdf:RDF>

```

In den Zeilen 1-3 wird wie üblich ein RDF-Dokument mit dem Tag `<rdf:RDF>` eingeleitet und die verwendeten Namensräume werden deklariert. In Zeile 3 wird auf die FOAF-Ontologie verwiesen, welche das für dieses Beispiel benötigte Vokabular zum Beschreiben von Personen und deren Beziehungen zur Verfügung stellt. In Zeile 4 beginnt eine Ressource vom Typ `foaf:Person`. Mit dem Attribut `rdf:about` wird die Ressource durch eine URI identifiziert. Das Beispiel nutzt eine relative URI, ausgehend von der des RDF-Dokuments. Wenn das Dokument zum Beispiel über `http://maxmuster.example.org/data/foaf.rdf` abrufbar ist, erhält die Ressource die die Person Max Muster darstellt folgende URI:

```
http://maxmuster.example.org/data/foaf.rdf#me
```

Über `<foaf:title>` wird in Zeile 5 die Anrede der Person bestimmt. In Zeile 6 wird der Name angegeben, der in den Zeilen 7 und 8 nochmals nach Vorname und Familienname aufgeschlüsselt wird. FOAF stellt auch ein Prädikat bereit um die E-Mail-Adresse einer Person anzugeben. Zum Schutz der persönlichen Daten und zur Abwehr von Spam gibt es mit `<foaf:mbox_sha1sum>` (Zeile 9) aber auch die Möglichkeit, nur den SHA1-Hash der Adresse zu hinterlegen. Der Hash ermöglicht es zu verifizieren ob man die korrekte E-Mail-Adresse der Person besitzt, ohne dass Dritte in den Besitz der Adresse kommen.¹

Dokumente wie Websites sind aus Sicht von Linked Data nichts anderes als Ressourcen und ebenso können sie behandelt werden. Deswegen verweist Zeile 10 über die URL im Attribut `rdf:resource` von `<foaf:website>` auf die Website der beschriebenen Person.

In den Zeilen 11 und 12 wird über `<foaf:knows>` eine Beziehung zwischen Max Muster und der Person aufgebaut, die über die URI `http://example.org/creator/franzmayer` identifiziert wird. Das Prädikat sagt aus, dass Max Muster diese Person kennt. Auch eingebettete Ressourcen sind möglich, wie die Zeilen 13-18 zeigen: Hier wird als Objekt der `<foaf:knows>`-Beziehung eine neue Person definiert.

¹In Listing 3.9 sind die Hashes zur besseren Lesbarkeit gekürzt

RDFS

Resource Description Framework Schema (RDFS) ist eine Beschreibungssprache für Ontologien. RDFS liegt ein Klassenkonzept zu Grunde, vergleichbar mit dem der objektorientierten Programmierung. In RDFS definierte Klassen können Ressourcen mit dem Prädikat `rdf:type` zugewiesen oder als Wurzelement verwendet werden. Dies wurde zum Beispiel in Listing 3.3 getan. Die dort verwendete Klasse `BibliographicResource` kann mittels RDFS wie in Listing 3.10 definiert werden.

Listing 3.10: Definition einer Klasse mittels RDFS

```
1 <rdf:RDF
2   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
4
5   <rdfs:Class
6     rdf:about="http://purl.org/dc/terms/BibliographicResource">
7     <rdfs:label>Bibliographic Resource</rdfs:label>
8     <rdfs:comment>
9       A book, article, or other documentary resource.
10    </rdfs:comment>
11    <rdfs:isDefinedBy rdf:resource="http://purl.org/dc/terms/">
12  </rdf:Description>
13
14 </rdf:RDF>
```

Das Dokument wird, wie jedes andere RDF/XML-Dokument auch, mit dem Wurzelement `<rdf:RDF>` eingeleitet. In diesem Fall wird lediglich noch der Namensraum von RDFS deklariert. Mittels `<rdfs:Class>` wird in Zeile 5 eine Klasse definiert, Zeile 6 gibt dieser Klasse eine Bezeichnung und in den Zeilen 7-9 wird in einem Kommentar angegeben, was diese Klasse bedeutet. Zeile 10 gibt das RDFS-Dokument an, in der die Klasse definiert ist.

Im Grunde genommen ist ein RDFS-Dokument ein herkömmliches RDF/XML-Dokument. `BibliographicResource` ist eine Ressource vom Typ `rdfs:Class` zu der Informationen in Form von Tripeln hinterlegt werden. Mit dem Prädikat `rdfs:subClassOf` können auch Unterklassen erzeugt werden.

Prädikate werden als Ressource des Typs `rdfs:Property` definiert. Listing 3.11 zeigt eine gekürzte Definition von `creator` aus Dublin Core.

Listing 3.11: Definition eines Prädikats mittels RDFS

```
1 <rdf:Property rdf:about="http://purl.org/dc/terms/creator">
2   <rdfs:label>Creator</rdfs:label>
3   <rdfs:comment>
```

```

4   An entity primarily responsible for making the resource .
5   </rdfs:comment>
6   <rdfs:isDefinedBy rdf:resource="http://purl.org/dc/terms/" />
7   <rdfs:range rdf:resource="http://purl.org/dc/terms/Agent" />
8   <rdfs:subPropertyOf rdf:resource="http://purl.org/dc/terms/
      contributor" />
9 </rdf:Property>

```

Zeile 1 leitet die Definition des Prädikats ein und gibt an, dass es über die URI `http://purl.org/dc/terms/creator` identifiziert wird. Als menschenlesbare Bezeichnung wird in Zeile 2 „Creator“ angegeben. Die Zeilen 3-5 enthalten wiederum einen Hinweis auf die Bedeutung des Prädikats und Zeile 6, wie schon in Listing 3.10, den Verweis auf das RDFS-Dokument. Neu ist `rdfs:range` in Zeile 7. Dadurch wird angegeben, welche Typen als Objekt genutzt werden können, wenn das definierte Prädikat in einem Tripel verwendet wird. In dem Beispiel wird festgelegt, dass als Verfasser nur Ressourcen vom Typ `http://purl.org/dc/terms/Agent` oder dessen Unterklassen in Frage kommen. Für Eigenschaften existiert ebenfalls eine Vererbungshierarchie: In Zeile 8 wird über `rdfs:subPropertyOf` angegeben, dass `dc:creator` eine Art von `dc:contributor` ist (Das ist jeder der einen Beitrag zu einer Ressource geleistet hat). Diese Information ist wichtig, wenn die Daten später durchsucht werden sollen. Angenommen es sollen alle Personen gefunden werden, die einen Beitrag zu einem Artikel geleistet haben. Dann dürfen nicht nur die Tripel berücksichtigt werden, die `dc:contributor` als Prädikat enthalten, sondern auch alle von diesem ererbenden Prädikate, wie `dc:creator`.

RDFS bietet darüber hinaus die Möglichkeit über `rdfs:domain` auch den Typ des Subjekts festzulegen, um die Nutzung des Prädikats auf sinnvolle Bereiche einzuschränken. Ein Prädikat `foo:preis` mag zum Beispiel für Produkte Sinn machen, ist jedoch bei der Beschreibung von Personen völlig fehl am Platze. Dieser Umstand kann durch die Verwendung von `rdfs:domain` formal ausgedrückt werden. Weitere Informationen zu RDFS bietet die Spezifikation des W3C[12].

OWL

Eine weitere Beschreibungssprache ist die Web Ontology Language. Die Abkürzung lautet tatsächlich OWL, der Buchstabendreher ist beabsichtigt². OWL basiert auf der Syntax von RDF, ist jedoch weitaus mächtiger in seinen Ausdrucksmöglichkeiten. Wie auch mit RDFS können Klassen und Prädikate definiert werden. Darüber hinaus sind jedoch auch prädikatenlogische Ausdrücke möglich. OWL bereitet damit das Fundament, um automatisiert logische Schlüsse aus den beschriebenen Daten zu ziehen und ist damit ein wichtiger Bestandteil des semantischen Webs. Aufgrund seiner Komplexität sei an dieser Stelle jedoch auf die W3C Recommendation verwiesen[13].

²<http://lists.w3.org/Archives/Public/www-webont-wg/2001Dec/0169.html>

3.3 Daten publizieren

In Abschnitt 3.2 wurde dargelegt, wie Informationen über Ressourcen und deren Beziehungen mit Hilfe von RDF und Ontologien formal beschrieben werden können. In diesem Abschnitt wird nun erörtert, wie diese Daten im Web veröffentlicht werden können. Es wird erläutert wie mit Hilfe der Inhaltsvereinbarung Informationen über Ressourcen auf den abfragenden Client zugeschnitten werden können. Weiterhin werden verschiedene Strategien vorgeschlagen um Ressourcen mit Hilfe von URIs zu benennen. Bei der Veröffentlichung von Linked Data ist außerdem die saubere Trennung zwischen Daten und Metadaten zu beachten, sowie die Unterscheidung zwischen reinen Informationsressourcen und solchen, die tatsächliche Dinge oder abstrakte Konzepte darstellen.

3.3.1 Arten von Ressourcen

In „Dereferencing HTTP URIs“ unterscheidet die W3C Technical Architecture Group zwischen Informationsressourcen und Nicht-Informationsressourcen [14]. Im WWW ist diese Unterscheidung von untergeordneter Relevanz, da alle Dokumente reine Informationsressourcen sind. Auch ein RDF-Dokument fällt in die Kategorie Informationsressource. Nicht-Informationsressourcen sind alle Dinge der realen Welt: Personen, Geräte, Medikamente, Pflanzen, Produkte und dergleichen. Darunter fallen auch abstrakte Dinge wie Themen, Diskussionen, Bestellungen und Ereignisse. Diese Unterscheidung ist im Kontext von Linked Data wichtig, da nun auch Dinge der realen Welt in Form von Nicht-Informationsressourcen im Web abrufbar werden sollen.

3.3.2 Inhaltsvereinbarung

Ein herkömmlicher Web-Browser kann nichts mit RDF anfangen, zeigt das rohe RDF/XML-Dokument oder bietet es zum Download an. Da dies wenig hilfreich ist, bietet es sich an mit der HTTP Inhaltsvereinbarung die Repräsentation der Daten auf den abfragenden Client zuzuschneiden. Das HTTP-Protokoll unterstützt ein Header-Feld mit dem ein Client den bevorzugten MIME-Type angeben kann. Abhängig von dieser Angabe kann ein Web-Server unterschiedliche Repräsentationen einer Ressource für unterschiedliche Clients ausliefern. Ein Web-Browser sendet üblicherweise `Accept: text/html` im Header der HTTP-Anfragen, während ein Client der an Linked Data interessiert ist den Header `Accept: application/rdf+xml` verwenden könnte.

Die Inhaltsvereinbarung für Nicht-Informationsressourcen kann mit Hilfe einer 303-Weiterleitung implementiert werden. Der HTTP-Statuscode 303 (See Other) bedeutet, dass sich die Antwort auf die vom Client gewünschte Anfrage an einer anderen Stelle befindet. Der Server sendet dann im Header-Feld `Location` die entsprechende Adresse.

Angenommen folgende URI identifiziert eine Nicht-Informationsressource, zum Beispiel ein Produkt:

1) <http://example.org/resource/product/0815>

Weiterhin existieren die folgenden zwei URIs, welche Informationsressourcen darstellen:

2) <http://example.org/data/product/0815>

3) <http://example.org/page/product/0815>

Nummer 1) identifiziert ein Objekt der realen Welt, also ein tatsächliches Produkt. Nummer 2) verweist auf eine Informationsressource, die das Produkt in Form von RDF/XML formal beschreibt und 3) auf eine HTML-Seite mit Informationen zu dem Produkt.

Fragt nun ein Client mit dem Header `Accept: text/html` nach der Ressource <http://example.org/resource/product/0815>, antwortet der Server mit einer 303-Weiterleitung auf <http://example.org/page/product/0815>. Verlangt der Client jedoch per `Accept: application/rdf+xml` nach RDF/XML, wird er auf die Informationsressource <http://example.org/data/product/0815> verwiesen. Auf diese Weise bekommt jeder Client die Repräsentation einer Nicht-Informationsressource, mit der er am meisten anfangen kann.

3.3.3 URI Namensstrategien

Wichtige Voraussetzung für ein Web aus verknüpften Daten ist, dass Ressourcen über eine URI eindeutig identifiziert werden können. Die Wahl der URI für eine Ressource sollte also wohlüberlegt sein. In „How to publish Linked Data on the Web“ [15] wurden bereits einige Ratschläge gegeben, die im folgenden aufgegriffen werden.

Die URI sollte sich zum einen gut als Namen eignen, der auch leicht für Menschen lesbar und verständlich ist. Zum anderen muss die Dereferenzierbarkeit gewährleistet sein. Es ist deshalb wichtig, dass die URI in einem Namensraum liegt, über den der Veröffentlicher der Daten selbst die Kontrolle hat. Des Weiteren sollten technische Details aus der URI herausgehalten werden. Dies erhöht sowohl die Lesbarkeit, als auch die Stabilität der URI. Wenn sich eine URI später ändern sollte, zerstört dies alle Links, die auf die Ressource verweisen. Deswegen ist die Stabilität ein wichtiges Kriterium für ein konsistentes Web aus Daten. Eine schlechte Wahl für eine URI wäre gemäß diesen Kriterien folgende:

<http://example.org/scripts/pageview.php?page=1&resource=5>

Es ist nicht erkennbar welches „Ding“ die URI identifiziert und technische Details sind erkennbar. Hinter der URI verbirgt sich nicht wirklich ein Ding der realen Welt, sondern ein PHP-Skript, das erst zusammen mit den übergeben Parametern eine Ressource darstellt. Es ist absehbar, dass diese URI irgendwann ungültig wird, oder etwas völlig anderes identifiziert.

Weiterhin ist zu beachten, dass für die Inhaltsvereinbarung (vgl. Abschnitt 3.3.2) in der Regel drei verschiedene URIs benötigt werden. Erstens für die Nicht-Informationsressource selbst, zweitens für die zugehörige RDF-Informationsressource und drittens eine eben solche, die von normalen Web-Browsern verstanden werden kann. Eine mögliche Benennung geht aus Abschnitt 3.3.2 hervor. Eine andere Möglichkeit wäre eine Unterscheidung nach Subdomain, zum Beispiel:

`http://example.org/product/0815` (Nicht-Informationsressource)

`http://data.example.org/product/0815` (RDF-Repräsentation)

`http://page.example.org/product/0815` (HTML-Repräsentation)

Es gibt noch eine weitere Strategie zur Benennung von Ressourcen, die auf Hash-URIs basiert. Diese Variante macht sich zu nutze, dass der Fragment-Teil (vgl. Abbildung 3.1) bei der Dereferenzierung einer URI nicht beachtet wird. Nicht-Informationsressourcen können also durch ein nachgestelltes # identifiziert werden, während der Rest der URI auf die RDF-Informationsressource verweist. Zum Beispiel könnte ein Produkt über folgende URI identifiziert werden:

`http://example.org/products.rdf#0815`

Vorteil dieser Variante ist, dass in `http://example.org/products.rdf` Informationen über mehrere Ressourcen der gleichen Familie (z.B. Produkte in diesem Fall) hinterlegt sein können und die einzelnen Nicht-Informationsressourcen durch eine per # abgetrennte Kennung identifiziert werden. Da keine 303-Weiterleitung benötigt wird, verringert sich die Anzahl an nötigen HTTP-Anfragen. Allerdings sieht die Variante in dieser Form auch keine HTML Informationsressource vor. Wie man Hash-URIs mit der Inhaltsvereinbarung kombinieren kann zeigt „Cool URIs for the Semantic Web“ [16]. Dort finden sich auch weitere Ratschläge für die geschickte Wahl von URIs im Kontext des Semantischen Webs.

3.3.4 Abgrenzung von Daten und Metadaten

Bei der Veröffentlichung von Linked Data ist sorgfältig zu beachten, was die eigentliche Ressource ist, die beschrieben wird, und wobei es sich lediglich um Informationen über diese Ressource handelt. Die Website über eine Produkt ist etwas anderes als das Produkt selbst. Trotzdem kann ein Produkt eine eindeutige, es identifizierende URI

haben. Wenn die Inhaltsvereinbarung korrekt umgesetzt wurde wird beim Aufruf dieser URI mit einem üblichen Web-Browser sogar die Website des Produkts angezeigt. Dies darf nicht zu der Verwechslung führen, dass die Website und das Produkt im Web das gleiche sind. Beides wird durch unterschiedliche URIs identifiziert. Ebenso ist die RDF-Repräsentation des Produkts vom Produkt selbst zu unterscheiden. Hinzu kommt, dass über die Website oder das RDF-Dokument selbst wieder Metadaten hinterlegt sein können, zum Beispiel wann das Dokument zuletzt geändert wurde und wer es geschrieben hat. Eine solche Information wäre innerhalb der Daten über das Produkt fehl am Platze. Ein Produkt hat in der Regel kein Änderungsdatum, eine Website oder ein RDF-Dokument hingegen schon. Bei der Veröffentlichung von Linked Data und der Verlinkung von Ressourcen ist es daher wichtig zu beachten, mit welcher Art von Ressource man es zu tun hat und ob tatsächlich Daten zu dieser Ressource veröffentlicht werden sollen oder Metadaten.

3.4 Daten konsumieren

In den Abschnitten 3.2 und 3.3 wurde aufgezeigt, wie Linked Data erstellt und im Web publiziert werden kann. Im Folgenden wird nun aufgezeigt, wie auf die Daten zugegriffen werden kann und wie sie genutzt werden können.

3.4.1 Linked Data Browser

Angelehnt an herkömmliche Web-Browser wurden bereits einige auf RDF und Linked Data spezialisierte Daten-Browser entwickelt. Ein Daten-Browser zeigt RDF-Dokumente nicht in ihrer Rohform an, sondern bereitet sie menschenlesbar auf, ähnlich wie dies ein Web-Browser mit HTML-Dokumenten macht.

Tim Berners-Lee hat mit „Tabulator“ einen Prototypen für einen solchen Browser entwickelt. Für Tabulator gibt es auch ein Plugin für den Browser Firefox³. Allerdings bleibt Firefox in erster Linie ein Web-Browser und fordert im Accept-Header HTML-Dokumente an. Mit Hilfe des Plugins „Modify Headers“⁴ kann der Header jedoch durch den Nutzer manipuliert werden, sodass die RDF-Variante einer Ressource angefordert wird. Tabulator ist eher ein Machbarkeitsnachweis, als ein für den Alltag geeigneter Datenbrowser, jedoch für erste Gehversuche mit Linked Data nach Erfahrungen des Autors gut geeignet.

³<http://dig.csail.mit.edu/2007/tab/>

⁴<https://addons.mozilla.org/en-US/firefox/addon/967>

Weitere Browser gibt es in Form eines Web-Frontends, zum Beispiel den OpenLink Data Explorer⁵, Disco⁶ oder Marbles⁷. Tabulator hat den Vorteil, dass verwandte Daten automatisch nachgeladen werden und dass verlinkte Ressourcen zu einer Baumstruktur aufgeklappt werden können. Abbildung 3.3 zeigt Tabulator beim Anzeigen der Ressource #me aus Listing 3.9.

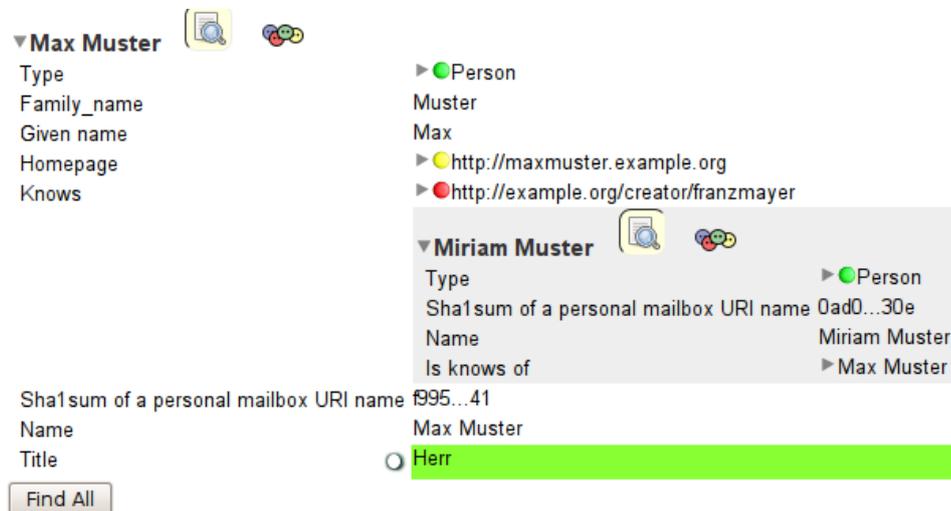


Abbildung 3.3: Der Daten-Browser Tabulator beim Anzeigen einer Ressource

3.4.2 SPARQL

SPARQL ist eine Sprache mit deren Hilfe Informationen aus einem RDF-Graphen abgefragt werden können. Ähnlich wie mit SQL Daten aus einer Datenbank erfragt werden können, ist dies mit Hilfe von SPARQL innerhalb eines RDF-Graphen möglich. Ein einfaches Beispiel für eine solche Abfrage zeigt Listing 3.12

Listing 3.12: Einfache SPARQL-Abfrage

```

1 PREFIX dc: <http://purl.org/dc/terms/>
2 SELECT ?title , ?date
3 WHERE {
4   ?x dc:title ?title .
5   ?x dc:modified ?date .
6   ?x dc:creator <http://example.org/creator/maxmuster> .
7 }

```

Die Abfrage wird in Zeile 1 zunächst durch die Definition eines Prefixes eingeleitet, um eine Kurzschreibweise der URIs zu ermöglichen. Der Rest der Abfrage ähnelt einem

⁵<http://demo.openlinksw.com/DAV/JS/rdfbrowser/index.html>

⁶http://www4.wiwiwiss.fu-berlin.de/rdf_browser/

⁷<http://www5.wiwiwiss.fu-berlin.de/marbles/>

SQL-Statement. Durch das `SELECT` in Zeile 2 wird angegeben, welche Informationen erfragt werden sollen. Mit einem Fragezeichen werden in SPARQL Variablen eingeleitet. Die Werte der Variablen `?title` und `?date` werden über die Bestimmungen im `WHERE`-Teil der Abfrage (Zeilen 3-7) ermittelt. Die Zeilen 4-6 geben dazu RDF-Tripel an, die erfüllt werden müssen. Die Tripel enthalten wiederum Variablen, die mit passenden Werten belegt werden, um die Anfrage zu erfüllen. Zeile 6 zum Beispiel trifft für alle Ressourcen zu, deren Verfasser Max Muster ist. Die Variable `?x` wird somit mit diesen Ressourcen belegt. Für alle jene Ressourcen `?x` wird über die Zeile 4 der Titel ermittelt und in der Variable `?title` festgehalten. Gleiches passiert für das Datum der letzten Änderung und die Variable `?date`. Die Abfrage bewirkt somit, dass der Titel und das Datum der letzten Änderung aller Ressourcen ausgegeben werden, die von Max Muster verfasst wurden.

Es fällt auf, dass der SQL-typische `FROM`-Teil fehlt. Dieser ist auch bei SPARQL möglich. Darüber kann der RDF-Graph bestimmt werden, auf den sich die Abfrage bezieht. In der Regel ist der `FROM`-Teil jedoch nicht nötig, da dem SPARQL-Server an den sich die Anfrage richtet, der zu Grunde liegende Graph bekannt sein sollte. Zudem liegt gerade die Stärke in einem Web aus Daten darin, Informationen zu finden, unabhängig davon wo sie liegen, allein dadurch, dass sie miteinander verknüpft sind. Die Beschränkung auf eine einzelne, abgeschottete RDF-Datenquelle ist dabei nicht zielführend. Die Ausführung von SPARQL-Anfragen über das gesamte Web aus Daten hinweg wird in „Executing SPARQL Queries over the Web of Linked Data“[17] ausführlich diskutiert. Nähere Informationen zur Syntax und Semantik von SPARQL liefert die W3C Recommendation[18].

3.4.3 Semantische Suchmaschinen

Semantische Suchmaschinen berücksichtigen neben der reinen Textinformation auch die Bedeutung von Informationen im Web. Während die Suchanfrage „Welcher Fluss fließt durch Karlsruhe?“ bei herkömmlichen Suchmaschinen lediglich eine Vielzahl von Dokumenten liefert, die eben diesen Text enthalten, sollen semantische Suchmaschinen die Bedeutung der Anfrage analysieren und passende Antworten geben. Schwierigkeiten in der Praxis entstehen dabei derzeit bei der Erkennung der Bedeutung von natürlichsprachlichen Sätzen und durch den Mangel an strukturierten, semantischen Informationen im Netz. Die Veröffentlichung von Linked Data ist deswegen eine wichtige Voraussetzung um semantische Suchmaschinen zukünftig möglich zu machen.

3.4.4 Data MashUps

Unter einem MashUp versteht man die Verknüpfung und Rekombination von vorhanden Inhalten zu etwas Neuem, ähnlich einer Collage in der Bildenden Kunst. Mit dem

Aufkommen des „Web 2.0“ wurden insbesondere multimediale Inhalte immer wieder neu verknüpft und in neue Zusammenhänge gestellt. Unter Nutzung von offenen Schnittstellen können zum Beispiel Karten aus Google Maps mit Bildern von der Foto-Community Flickr und eigenen Blogartikeln oder den Kurzmeldungen von Twitter verknüpft werden. Die Schaffung des Neuen liegt dabei nicht im Erstellen der Inhalte, sondern in deren Rearrangement.

Mit offenen, verknüpften Daten sind auch „Data MashUps“ denkbar, das heißt existierende Daten werden zusammengeführt, aufbereitet, neu kombiniert und in neue Zusammenhänge gestellt. Durch den neuen Blickwinkel auf die Daten können sich dadurch auch neue, unbekannte Informationen und Erkenntnisse auftun. Data MashUps sind jedoch kaum möglich, wenn Daten in unterschiedlichen oder gar unstrukturierten Formaten vorliegen. Linked Data bietet durch klar strukturierte Daten über Ressourcen, die durch URIs eindeutig identifiziert und abrufbar sind, die Voraussetzung um Data MashUps möglich zu machen.

Einen ersten Ansatz dieser Art bietet SIG.MA⁸ (Semantic Information Mashup). Mit SIG.MA lassen sich Daten aus unterschiedlichen Quellen zu einem Mash-Up zusammenführen. Die Schwierigkeit liegt hierbei nicht in der technischen Umsetzung, sondern am Mangel an Linked Data im Web.

⁸<http://sig.ma/>

4 Linked Data in der Praxis

Anhand von einigen Projekten und Anwendungen wird in diesem Kapitel verdeutlicht, wie Linked Data bereits in der Praxis eingesetzt wird.

4.1 DBpedia

Eines der größten Projekte unter Einsatz von Linked Data ist die DBpedia¹. Die DBpedia extrahiert Daten aus der Wikipedia und stellt sie in strukturierter Form zur Verfügung. DBpedia stellt für jeden Wikipedia-Artikel eine Ressource zur Verfügung, die über eine URI der folgenden Form identifiziert wird:

```
http://dbpedia.org/resource/<Artikelname>
```

DBpedia macht dabei von der Inhaltsvereinbarung (siehe Abschnitt 3.3.2) Gebrauch, sodass eine HTML-Variante unter <http://dbpedia.org/page/<Artikelname>> verfügbar ist und die RDF-Daten unter <http://dbpedia.org/data/<Artikelname>> abgerufen werden können.

Dadurch dass DBpedia auf die Wikipedia zurückgreift, ist sie eine der größten Quellen an Linked Data im Web und wird oft von anderen Daten verlinkt. Doch genau genommen ist das Vorgehen der DBpedia falsch, denn es werden Daten aus für Menschen bestimmten HTML-Seiten abgegriffen. Eigentlich sollte es umgekehrt sein: Grundlage für die Wikipedia sollten strukturierte Daten sein, die dann sowohl als RDF als auch in Form von menschenlesbaren HTML-Seiten verfügbar gemacht werden.

Leider geht die Wikipedia diesen Schritt aktuell nicht. DBpedia kann daher als ein legitimes Behelfsmittel betrachtet werden, um große Mengen an Linked Data zu erhalten. Zukünftig sollte es jedoch selbstverständlich werden, dass Linked Data direkt an der Quelle entsteht und nicht erst über den Umweg des Auslesens von HTML-Seiten oder anderen unzugänglichen Datenquellen.

DBpedia stellt auch eine SPARQL-Schnittstelle bereit, um die Daten gezielt über mehrere Ressourcen hinweg abzufragen. Einige Beispielabfragen stehen zur Verfügung mit denen es sich zu experimentieren lohnt², zum Beispiel „Alle Fußballspieler die als Torwart für einen Verein gespielt haben, der ein Stadion mit mehr als 40.000 Sitzen

¹<http://dbpedia.org>

²<http://wiki.dbpedia.org/OnlineAccess#h28-5>

besitzt und die in einem Land mit mehr als 10 Millionen Einwohnern geboren wurden.“ Wie lange es dauert diese Informationen manuell aus der Wikipedia heraus zu suchen, möge der Leser selbst beurteilen.

4.2 Faviki

Faviki³ ist ein sogenannter Social-Bookmarking-Dienst. Benutzer können bei einem solchen Dienst ihre Online-Lesezeichen zentral im Netz speichern, statt auf ihrem lokalen Browser. Zusätzlich können die Lesezeichen auch veröffentlicht und so mit anderen geteilt werden. Faviki nutzt semantische Tags zur Kategorisierung der Lesezeichen. Das Problem beim herkömmlichen Tagging ist, dass Begriffe zweideutig sind und somit Lesezeichen miteinander in Verbindung gebracht werden, die eigentlich nichts miteinander zu tun haben. Bei einer Website die mit „Jaguar“ getaggt wird, kann es entweder um Raubtiere oder um Autos gehen. Faviki umgeht dieses Problem, indem es eindeutig identifizierbare Ressourcen als Tags verwendet, z.B. <http://dbpedia.org/resource/Jaguar> für das Raubtier und http://dbpedia.org/resource/Jaguar_Cars für die Automobilmarke. Dies verhindert auch das Problem, dass Nutzer unterschiedliche Schreibweisen für den gleichen Sachverhalt verwenden. Wenn ein Nutzer ein Tag eingibt verbindet sich der Dienst mit der DBpedia und schlägt passende Ressourcen vor, aus denen der Nutzer die richtige auswählt. Die Bezeichnung des Tags ist frei wählbar, die Bedeutung ist jedoch eindeutig durch die ausgewählte Ressource festgelegt. Durch die hinterlegte Bedeutung kann der Dienst auch sehr treffend verwandte Themen lokalisieren und weiterführende Informationen anbieten.

4.3 revyu.com

Im Web gibt es bereits zahlreiche Seiten, bei denen man Dinge bewerten kann: Den letzten Urlaub, Bücher, Filme, Restaurants - für fast alles gibt es eine spezialisierte Seite für Bewertungen und Rezensionen. Die Seite revyu.com geht einen Schritt weiter und macht dabei Gebrauch von Linked Data und der Idee alles über URIs zu identifizieren. Auf revyu.com lassen sich Bewertungen zu allem verfassen, was irgendwie benannt werden kann. Sowohl der Bewertungsgegenstand, als auch die Bewertung selbst und Informationen über deren Autor werden dabei als Linked Data in Form von RDF online gestellt.

Während revyu.com damit einen wichtigen Schritt geht um das Web aus Daten zu bereichern, vertut es auch einige Chancen: Zum Beispiel ist es bisher nicht möglich externe Ressourcen zu bewerten. Entweder hat ein Bewertungsgegenstand bereits eine

³<http://faviki.com>

URI bei revyu.com und eine neue Bewertung wird diesem hinzugefügt, oder die Ressource wird neu bei dem Dienst angelegt. Dem Gedanken von Linked Data folgend sollte es auch möglich sein, eine beliebige URI anzugeben und das durch sie identifizierte „Ding“ zu bewerten. Hinzu kommt dass alle Bewertungsgegenstände den Typ **Thing** haben und nicht etwa einen passenden konkreteren Typ, wie Buch, Film oder ähnliches.

Alles in allem geht revyu.com jedoch einen Schritt in die richtige Richtung und könnte sich mit den genannten Nachbesserungen zu einer wertvollen Linked-Data-Plattform entwickeln.

4.4 Triplify

Mit Hilfe von Triplify⁴ können Inhalte relationaler Datenbanken als Linked Data im Web verfügbar gemacht werden. Viele Web-Anwendungen, wie Blogs, Wikis, Webshops und Foren, basieren auf Datenbanken. In diesen Datenbanken steckt in der Regel mehr Semantik als in Form der Webseiten die daraus generiert werden veröffentlicht wird. Triplify will diese Lücke schließen und die Semantik der Datenbank als Linked Data verfügbar machen.

Triplify muss dazu für eine konkrete Web-Anwendung bzw. die zugrunde liegende Datenbank konfiguriert werden. Dazu müssen SQL-Statements definiert werden, die die gewünschten Informationen aus der Datenbank auslesen und bestimmten Teilen eines RDF-Tripels zuweisen. Vorgefertigte Konfigurationen gibt es unter anderem für die Blog-Software Wordpress und das Content-Management-System Joomla. Durch die Verfügbarkeit für solch weit verbreitet Web-Anwendungen und die leichte Konfigurierbarkeit für andere Applikationen, bietet Triplify die Chance, Linked Data großflächig im Web zu verteilen. Wie aber auch die DBpedia ist Triplify eher ein Behelfsmittel. Wünschenswert wäre, dass Web-Applikationen unmittelbar die Veröffentlichung ihrer Daten als Linked Data möglich machen.

⁴<http://triplify.org/>

5 Chancen von Linked Data

Nachdem in Kapitel 3 die Techniken hinter Linked Data erläutert und in Kapitel 4 einige praktische Anwendungen vorgestellt wurden, beleuchtet dieses Kapitel die Chancen von Linked Data. Das Kapitel betrachtet sowohl die wirtschaftlichen Chancen, als auch den Nutzen für Staat und Gesellschaft sowie Wissenschaft und Forschung. Abschließend wird auf mögliche Gefahren und Risiken von Linked Data eingegangen.

5.1 Linked Data aus wirtschaftlicher Sicht

Als das World Wide Web entwickelt wurde, lag der Fokus auf der Zusammenarbeit und dem Informationsaustausch unter Wissenschaftlern. Dass die weltweite Vernetzung von Dokumenten auch ein wichtiger Marktplatz werden würde, war damals noch nicht absehbar. Im Rahmen dieses Kapitels wird erörtert, wie Linked Data einen wirtschaftlichen Nutzen entfalten kann. Es wird darauf eingegangen, welche Chancen offene und verknüpfte Daten für den Verbraucher bieten und wie Unternehmen von Linked Data profitieren können.

5.1.1 Linked Data als Chance für Verbraucher

Markttransparenz gilt als Grundlage für einen vollkommenen Markt, bei dem alle Marktteilnehmer über sämtliche Gegebenheiten des Marktes umfassend informiert sind und die Folgen ihres Handelns abschätzen können. Dazu gehört insbesondere die Kenntnis aller Preise, Güter, Angebote und Konditionen. Vollständige Markttransparenz kann nie erreicht werden, je mehr Informationen jedoch über einen Markt vorliegen, als desto transparenter gilt er.

Linked Data kann dazu beitragen, den Kunden und Verbrauchern einen besseren Überblick über Waren und Dienstleistungen zu verschaffen und die Vergleichbarkeit von Angeboten zu erhöhen. Die so geschaffene Transparenz führt zu einer sachlicheren Gegenüberstellung von konkurrierenden Marktteilnehmern und erhöht dadurch den Wettbewerb. Die Verbraucher profitieren durch sinkende Preise und sind besser informiert.

Ein Beispiel aus dem Mobilfunkmarkt: Angenommen alle Angebote und Konditionen sind durch Linked Data beschrieben und veröffentlicht. Der Verbraucher kann nun auf

einheitlicher Basis alle Mobilfunktarife miteinander vergleichen und den für ihn am besten geeigneten auswählen. Versteckte Kosten treten nicht auf, da alle Informationen strukturiert vorliegen. Der Verbraucher könnte zudem eine Software zur Hilfe nehmen, die auf den Vergleich von Mobilfunkangeboten spezialisiert ist. Mit dieser Software könnte er die gewünschten Tarifoptionen angeben, sowie die maximalen Kosten festlegen und so gezielt passende Angebote aus allen im Web verfügbaren Informationen herausfiltern um seine Wahl zu treffen.

5.1.2 Linked Data als Chance für Unternehmen

Der Bedarf, umfassend über den Markt informiert zu sein, existiert auch auf Anbieterseite. Unternehmen versuchen heute zum Beispiel mittels Marktforschung über die Gegebenheiten am Markt informiert zu sein. Der Informationsbedarf erstreckt sich sowohl auf die Stellung des eigenen Unternehmens, als auch auf die Situation der Konkurrenz und den Bedarf der Verbraucher. Zur Kalkulation der Preise ist es zum Beispiel wichtig zu wissen, wie viel Geld die Kunden für welche Qualität zu zahlen bereit sind und wann sie zur Konkurrenz abwandern.

Wenn ein Unternehmen Daten über seine Produkte und Angebote in Form von Linked Data ins Netz stellt, profitieren nicht nur die Kunden, sondern auch das Unternehmen, wenn es die Daten zu nutzen und auszuwerten weiß. Angenommen ein Mobilfunkanbieter macht seine Tarifinformationen als Linked Data verfügbar. Gegenüber der Konkurrenz hat er nun den Wettbewerbsvorteil, dass seine Angebote leichter auffindbar und transparent sind. Sofern die Konkurrenz diesen Schritt noch nicht gegangen ist, mangelt es noch an der Vergleichbarkeit der Angebote, was sich für den Anbieter kurzzeitig nachteilig auswirken kann, nämlich wenn die Verbraucher durch mangelnde Information die Konkurrenzangebote vorziehen. Es ist jedoch zu erwarten, dass dies mittelfristig zu größerer Unzufriedenheit mit den Konkurrenzprodukten führt und die Verbraucher zukünftig transparente Angebote vorziehen. Die Konkurrenzanbieter gelangen dadurch in Zugzwang und sind darauf angewiesen ihre Daten ebenfalls verfügbar zu machen. Der Anbieter der frühzeitig mit Linked Data am Markt war profitiert durch seine längere Erfahrung und den Vertrauensbonus der Kunden.

Darüber hinaus kann der Anbieter die im Web verfügbaren Daten für Marktforschung und Data Mining nutzen. Kunden schreiben schon heute zahlreiche Blogartikel und Bewertungen über Waren und Dienstleistungen. Sind diese durch Linked Data tatsächlich mit den Produkten des Unternehmens verknüpft, erleichtert dies die Auswertung der Informationen und die sich daraus ergebende Wertschöpfung. Ein Unternehmen kann so leicht herausfinden, wie ein Produkt am Markt ankommt, wie es von welchen Kundentypen bewertet wird und worüber diese Kunden noch schreiben. Diese Informationen stecken bereits heute im Web, lassen sich jedoch nur mit hohem Aufwand

herausfinden. Ein Web aus Daten wird die Informationsgewinnung für Unternehmen deutlich erleichtern.

Gleichzeitig treten auch Unternehmen im Markt als Kunden auf, zum Beispiel gegenüber ihren Lieferanten, und profitieren ebenso wie der Endverbraucher von den im vorherigen Abschnitt genannten Vorteilen.

5.2 Wissenschaft und Forschung

Das WWW hat den sekundenschnellen Austausch von wissenschaftlichen Dokumenten möglich gemacht. Mit Hilfe von Linked Data ist es möglich, Rohdaten ebenso verfügbar zu machen. Daten müssen somit nicht immer wieder neu erhoben werden, sondern Forscher können auf einen gemeinsamen Datenbestand zurückgreifen und ihn auf unterschiedliche Arten auswerten. Durch die Verknüpfung mehrerer Datenbestände können zudem neue wissenschaftliche Erkenntnisse gewonnen werden, die ohne verknüpfte Daten kaum auffindbar wären. Ein Beispiel um diese Chance zu verdeutlichen: Ein Mediziner veröffentlicht Daten über verschiedene Stoffe und deren heilende Wirkung. Gleichzeitig stellt ein Botaniker Informationen zur Verfügung, welche Stoffe von ihm entdeckte Pflanzen enthalten. Über eine Suche im Web entdeckt der Mediziner nun, dass eine dieser Pflanzen zur Herstellung eines der Heilmittel verwendet werden kann. Weder der Botaniker, noch der Mediziner wären alleine, ohne die Verknüpfung der Daten auf diesen Sachverhalt gestoßen. Es ist davon auszugehen, dass die Datenbestände die der Wissenschaft aktuell vorliegen und in Zukunft entstehen werden ein großes Potential an versteckten Informationen enthalten, die nur durch die Verknüpfung dieser Daten entdeckt werden können.

5.3 Demokratie und Staatswesen

Der freie Zugang zu staatlichen Informationen ist für eine demokratische Informationsgesellschaft wichtiger denn je. Die Informationsfreiheitsgesetze des Bundes und der Länder haben einen Anspruch der Bürger auf staatliche Informationen gesetzlich festgeschrieben. Allerdings nutzt der Staat die Chancen die das Internet und die Informationstechnologie für die Veröffentlichung und Verbreitung von Informationen bieten kaum. Daten werden oft nur in vorgefertigten Statistiken veröffentlicht und als statische Webseiten oder PDF-Dokumente ausgeliefert. Eine Chance über offene Schnittstellen auf Rohdaten zuzugreifen und diese selbst zu verarbeiten und in neue Zusammenhänge zu setzen hat der Bürger in der Regel nicht.

Die amerikanische Regierung ist mit dem Projekt [data.gov](http://www.data.gov/)¹ erstmals in diese Richtung gegangen, vertut jedoch einige Chancen dadurch, dass den Daten die Semantik fehlt und sie nicht untereinander verknüpft sind. Die Datenbestände stehen zum Beispiel als XML- oder CSV-Dateien bereit. Dies ist zwar ein Fortschritt gegenüber dem schwer zu verarbeitenden PDF, alle Chancen die sich durch die weltweite Verknüpfung der Daten ergeben bleiben jedoch ungenutzt.

Ein Staat der seine Daten als Linked Data veröffentlicht, gibt ein Stück weit Kontrolle zurück an die Bürger. Dies hindert den Staat nicht daran, weiter selbst Arbeitslosenstatistiken und demografische Informationen zu veröffentlichen. Dem Bürger wird jedoch die Macht gegeben diese Informationen anhand der Rohdaten zu kontrollieren, aus anderen Blickwinkeln zu betrachten und neu zu bewerten. Natürlich ist nicht jedem Bürger zuzumuten, dies selbst zu tun. Aber die Möglichkeit macht den Staat ein Stück weit demokratischer. Zudem gibt es zahlreiche Nicht-Regierungsorganisationen und private Projekte die die Daten zum Nutzen aller weiterverarbeiten können. Als ein Beispiel sei das Projekt [Abgeordnetenwatch](http://www.abgeordnetenwatch.de/)² genannt, welches unter anderem auch die Ergebnisse aller namentlichen Abstimmungen des Bundestages veröffentlicht und durchsuchbar macht. Das Projekt ist derzeit umständlich darauf angewiesen, die Daten aus einem vom Bundestag bereitgestellten PDF-Dokument auszulesen. Würden die Ergebnisse als Linked Data bereit stehen, würde nicht nur Abgeordnetenwatch davon profitieren, sondern es könnten zahlreiche Stellen im Netz auf die Daten verweisen, sie für Data MashUps (siehe Abschnitt 3.4.4) verwenden, mit weiteren Informationen verknüpfen und so mehr Transparenz in die Politik bringen.

5.4 Gefahren und Risiken von Linked Data

Bei allen Vorteilen von Linked Data dürfen auch einige Gefahren und Risiken der strukturierten Verknüpfung von Daten nicht außer Acht gelassen werden. Zum einen ist der Datenschutz und die informationelle Selbstbestimmung von Personen gefährdet. Der Datenschutz hat das Ziel, personenbezogene Daten vor Missbrauch zu schützen. Personenbezogene Daten sind alle Daten, die unmittelbar oder mittelbar einer natürlichen Person zugeordnet werden können. Der Datenschutz soll das Recht auf informationelle Selbstbestimmung gewährleisten. Dieses Grundrecht wurde vom Bundesverfassungsgericht aus der Würde des Menschen (Art. 1 Grundgesetz) und dem allgemeinen Persönlichkeitsrecht (Art. 2 Grundgesetz) abgeleitet. Es gesteht dem Einzelnen das Recht zu, selbst über die Verwendung und Veröffentlichung seiner personenbezogenen Daten zu entscheiden.

¹<http://www.data.gov/>

²<http://www.abgeordnetenwatch.de/>

Aus diesen Rechten folgt, dass jede Veröffentlichung von personenbezogenen Daten im Web nur mit Zustimmung des Betroffenen erlaubt ist. Mit FOAF wurde in dieser Studienarbeit bereits eine Ontologie vorgestellt, die darauf spezialisiert ist, Informationen über Personen und deren Beziehungen zu veröffentlichen. Es ist nur eine Frage der Zeit, bis auf diese Weise Informationen auch missbräuchlich, ohne Kenntnis oder gar zum Schaden des Betroffenen, veröffentlicht werden. Allerdings besteht das Problem des Missbrauchs bereits derzeit im Netz. Die Verknüpfung und Struktur von Linked Data könnte insofern den Datenschutz sogar stärken, da Betroffene missbräuchlich veröffentlichte Daten leichter aufspüren und die Ursache bekämpfen könnten. Trotzdem sollte die Veröffentlichung von Linked Data auch stets unter Datenschutzaspekten kritisch beobachtet werden. Gleichzeitig sollte sich jeder Nutzer Gedanken machen, inwieweit er persönliche Informationen tatsächlich im Web veröffentlichen möchte und auch mit Linked Data sparsam umgehen.

Ein weiteres Problem sind Falschinformationen und Widersprüche im Web aus Daten. Das aktuelle Web erfordert bereits eine hohe Medienkompetenz und ein kritisches Denken um die Fülle an Informationen richtig einschätzen zu können. Dies wird sich auch im Web aus Daten nicht ohne Weiteres ändern. Die Gefahr ist, dass strukturierte Daten eine höhere Sachlichkeit und Glaubwürdigkeit suggerieren, diese aber nicht zwangsläufig gegeben ist. Linked Data muss genauso kritisch begutachtet werden, wie alle anderen Informationen im Web. Mit der Frage des Vertrauens und der Glaubwürdigkeit befasst sich im semantischen Web die Trust-Ebene (vgl. Abbildung 2.1 auf Seite 7), die jedoch noch nicht praktisch umgesetzt ist.

6 Zukunftsaussichten

Dieses Kapitel wirft einen Blick auf die Zukunft von Linked Data. In den vorangegangenen Kapiteln wurden die technischen Grundlagen erklärt, ein Blick auf die aktuellen praktischen Einsatzmöglichkeiten geworfen und die Chancen von Linked Data aufgezeigt. An dieser Stelle wird nun darauf eingegangen welche weiteren Schritte für die Etablierung eines Webs aus Daten nötig sind und was die Vision eines solchen Webs sein könnte.

6.1 Problem „Database Hugging“

Damit das Web aus Daten wächst, ist ein Umdenken erforderlich. Denn unzählige Projekte im Web ziehen die Veröffentlichung einer hübsch gestalteten Website noch der Veröffentlichung von Rohdaten vor. Wertvolle Informationen liegen teilweise sogar nur in Form von Flash-Anwendungen vor. Solch ansprechend gestaltete Informationsportale mögen noch so lehrreich sein: Die Verwendung des dort hinterlegten Wissens ist stark eingeschränkt. Die Informationen sind regelrecht hinter solchen Websites gefangen und können nicht darüber hinaus verwendet werden.

Dieses Problem umschrieb Tim Berners-Lee mit dem Begriff „Database Hugging“, der Umklammerung von Datenbanken. Anbieter von Informationen im Web sitzen regelrecht auf ihren Datenbanken und geben sie nur in aufbereiteter Form preis, einer Form die zwar menschenlesbar, aber maschinell schwer weiter zu verarbeiten ist. Es muss ein Umdenken stattfinden, dass die Rohdaten benötigt werden. Die Anbieter sollten die Rohdaten in Form von Linked Data veröffentlichen. Parallel dazu kann zusätzlich trotzdem noch eine ansprechende Website erstellt werden. Die Daten jedoch sind dann verfügbar und können auf viele andere Weisen ausgewertet und verknüpft werden.

6.2 Social Semantic Web

Das Web ist heute weit mehr als ein Informations- und Kommunikationsmedium. Es ist zu einem Ort geworden, in dem Menschen ihre sozialen Kontakte pflegen, künstlerisch tätig sind, an dem sie ihre Erlebnisse in Text, Bild und Ton festhalten und untereinander austauschen. Im sogenannten „Web 2.0“, oder „Social Web“, verwischen die Grenzen zwischen Anbietern und Nutzern. Es gibt nicht mehr denjenigen, der eine Website

betreibt und diejenigen die sie lesen. Websites sind interaktiv, Nutzer steuern eigene Inhalte bei. Immer öfter stehen die Inhalte der Nutzer und die Nutzer selbst sogar im Vordergrund.

Die Idee des „Social Semantic Web“ verknüpft die Ansätze der nutzergenerierten Inhalte mit Linked Data und dem Semantischen Web. Nutzer tragen bereits heute aktiv dazu bei die Informationen im Web zu kategorisieren und strukturieren, indem sie Inhalte „taggen“, das heißt mit Schlagworten versehen, sie in Gruppen und Kategorien einordnen, bewerten und filtern. Dies alles passiert „nebenbei“, weil es nötig ist um die Übersicht über die Informationsflut im Web zu erhalten. Für den Schritt ins „Social Semantic Web“ ist es erforderlich, dass die Ergebnisse dieser Tätigkeiten in Form strukturierter, verknüpfter Daten festgehalten werden. Ein praktisches Beispiel, wie dies möglich ist, liefert schon heute der Dienst Faviki (siehe Abschnitt 4.2) mit den semantischen Tags. Web-2.0 -Anwendungen, wie Foto- und Video-Plattformen, Soziale Netzwerke, Blogs, Lesezeichendienste und Wikis, sollten ebenfalls die manuelle Kategorisierung durch die Nutzer mit der Anreicherung von semantischen Informationen verknüpfen, um das Web aus Daten weiter auszubauen. In „Social Semantic Web. Web 2.0 - was nun?“[19] werden dazu Anwendungsmöglichkeiten und Perspektiven aufgezeigt.

6.3 Anwendungen und Tools

Im Umfeld von Linked Data gibt es bereits eine Vielzahl von Tools und Anwendungen, einige davon wurden im Rahmen dieser Studienarbeit vorgestellt. Oft handelt es sich jedoch mehr um Machbarkeitsbeweise und Prototypen, als um in der Praxis sinnvoll einsetzbare Software. Für die Zukunft von Linked Data ist es deshalb nötig, dass nicht nur immer mehr Daten in dieser Form veröffentlicht werden, sondern dass auch die Tools ihren Kinderschuhen entwachsen und zu professionellen Anwendungen werden.

Diese beiden Entwicklungen gehen Hand in Hand: Bessere Tools führen zu einem breiteren Interesse an Linked Data und machen die Erstellung und Veröffentlichung der Daten einfacher. Die dadurch entstehende Datenmenge macht die Programme wiederum zu noch nützlicheren Werkzeugen, da mehr Informationen ausgewertet werden und mit ihnen neue Zusammenhänge entdeckt werden können.

Denkbar sind semantische Wikis, die genauso leicht zu bedienen sind, wie derzeit schon, jedoch zusätzlich ihre Informationen als Linked Data verfügbar machen. Das gleiche gilt für Content-Management-Systeme jeder Art. Die Produktion von Linked Data muss so leicht gemacht werden, wie ein Kommentar in einem Blog.

Gleichzeitig sollten benutzerfreundliche Daten-Browser das durchforsten der Datenmengen auch für Normalbenutzer möglich machen. Man stelle sich einen Daten-Browser vor, der die Daten und deren Verknüpfungen ansprechend visualisiert, mit dem man per

Mausklick neue Verknüpfungen herstellen oder SPARQL-Anfragen visuell zusammen stellen kann: Das Web aus Daten würde zu einem Quell an leicht erschließbarem Wissen.

6.4 Vision

Die Vision von Linked Data ist ein Web als große, globale Datenbank. Ebenso wenig, wie das derzeitige Web Grenzen kennt, wie dort alle Dokumente untereinander verlinkt sind und es keine abgeschotteten „Inseln“ im Web gibt, sollen auch Daten zu einem einzigen globalen Web zusammenwachsen. Dieses Web wäre eine riesige, dezentrale, weltweite Datenbank, aus der alle Informationen automatisiert abgefragt werden können.

Durch die umfassende logische Verknüpfung riesiger Datenmengen entsteht jedoch weit mehr als nur eine Datensammlung. Mit Hilfe inferenzieller Methoden, lassen sich aus den Daten logische Schlüsse ziehen und so neues Wissen generieren. Das Web könnte zu einem Expertensystem heranwachsen, welches nicht, wie aktuelle Systeme, auf bestimmte Wissensgebiete beschränkt ist, sondern auf nahezu den gesamten Erfahrungsschatz der Menschheit zurückgreifen kann.

Wie sich das Web zukünftig tatsächlich entwickeln wird, lässt sich nur schwer erahnen. Doch jeder kann einen Beitrag zu dieser Entwicklung leisten. So wie das WWW aus den unzähligen Webseiten gewachsen ist, die jeder Nutzer online stellte, kann heute jeder einen Beitrag durch die Veröffentlichung von Daten leisten. Web-Entwickler können ihre Anwendungen um die Generierung von Linked Data erweitern, Forscher können ihre Rohdaten in Form von RDF online stellen und Blogger ihre Artikel mit Metadaten anreichern. Mit der Zeit wird das Web aus Daten wachsen und Anwendungsgebiete eröffnen, an die vermutlich heute noch niemand denkt.

7 Zusammenfassung und Fazit

In der vorliegenden Studienarbeit wurde das Konzept „Linked Data“ behandelt. Dabei wurden sowohl die technischen Grundlagen beschrieben, als auch praktische Einsatzmöglichkeiten und Chancen aufgezeigt.

Der Wandel vom „Web aus Dokumenten“ zum „Web aus Daten“ wurde verdeutlicht: Während das bestehende WWW aus untereinander verlinkten Dokumenten besteht, hat Linked Data das Ziel, Daten auf der Ebene ihrer Bedeutung untereinander zu verknüpfen, um Informationsmassen automatisiert auswerten zu können. Linked Data stellt damit einen Bestandteil des Semantischen Webs dar. Während es bei letzterem jedoch auch um die wissenschaftliche Formulierung von Wissen und Logik und die Verifikation von Informationen geht, befasst sich Linked Data auf einem niedrigeren Level mit der strukturierten Veröffentlichung und Verknüpfung von Daten.

Die Studienarbeit hat die technischen Grundlagen von Linked Data vermittelt. Die vier von Tim Berners-Lee verfassten Grundprinzipien wurden beschrieben. Diese sagen aus, dass nicht nur Dokumente, sondern auch alle möglichen Dinge im Web über eine URI identifiziert werden sollen, welche über das HTTP-Protokoll dereferenziert werden kann. Beim Abruf einer solchen Ressource sollen strukturierte Informationen angeboten werden, die mit anderen Ressourcen verknüpft sind.

Zur Erstellung von strukturierten und verknüpften Daten wurden in dieser Studienarbeit das Resource Description Framework und verschiedene Notationsweisen dieses Modells vorgestellt. Ebenso wurde der Begriff Ontologie eingeführt und mit Dublin Core und FOAF wurden zwei verbreitete Vokabulare vorgestellt. Als Vertreter der Ontologie-Beschreibungssprachen ging die Studienarbeit auf RDFS und OWL ein.

Die Unterscheidung zwischen Informationsressourcen und Nicht-Informationsressourcen war wichtig für die Veröffentlichung von Linked Data im Web. Die Studienarbeit erklärte, wie mit Hilfe der Inhaltsvereinbarung beim Zugriff auf eine Nicht-Informationsressource die zum abfragenden Client passende Informationsressource zur Verfügung gestellt werden kann. Zudem wurden verschiedene Strategien diskutiert, um Ressourcen sinnvoll zu benennen und die Abgrenzung von Daten und Metadaten wurde verdeutlicht.

Um Daten zu konsumieren hat die Studienarbeit verschiedene Möglichkeiten vorgestellt: Die Daten können zum Beispiel mit einem Data Browser angezeigt und durchstöbert werden. Mit SPARQL gibt es darüber hinaus eine Möglichkeit, Informationen mit Hilfe formaler Abfragen aus RDF-Graphen zu ermitteln. Semantische Suchmaschinen

sind noch nicht praxistauglich, jedoch stellt Linked Data eine wichtige Voraussetzung für deren Funktionieren dar. Ein weiteres Anwendungsfeld von Linked Data bieten Data MashUps, also Rekombinationen von vorhandenen Daten unter neuen Gesichtspunkten und in vorher nicht betrachteten Zusammenstellungen.

Praktische Anwendung findet Linked Data schon heute in Projekten wie der DBpedia, Faviki, revyu.com und Triplify. Die Studienarbeit hat jedoch aufgezeigt, dass diese Projekte noch Verbesserungspotential bieten oder ein Behelfsmittel auf dem Weg zu einem umfassenden Web aus Daten darstellen.

Die Studienarbeit beleuchtete auch die Chancen von Linked Data. Produkt- und Tarifinformationen in Form von Linked Data können den Markt für Verbraucher transparenter machen, indem zum Beispiel Angebotsvergleiche erleichtert werden. Ebenso bietet Linked Data Vorteile für Unternehmen, da die Informationsgewinnung über die eigene und fremde Marktposition deutlich erleichtert würde. Wissenschaft und Forschung können durch den Zugriff auf einen gemeinsamen, großen Datenbestand profitieren und durch die Verknüpfung unerwartete Zusammenhänge entdecken. Für Demokratie und Staatswesen bietet Linked Data die Chance, staatliche Informationen kostengünstig und für alle Bürger zugänglich zu veröffentlichen. Sowohl interessierte Bürger, als auch Nicht-Regierungsorganisationen könnten die Daten für ihre Projekte nutzen und die Politik ein Stück weit transparenter machen.

Gleichzeitig wurde auch auf mögliche Gefahren von Linked Data hingewiesen. Zum einen besteht eine Gefahr für den Datenschutz und die informationelle Selbstbestimmung bei der Veröffentlichung von personenbezogenen Daten. Ein Missbrauch ist sowohl möglich, wenn Daten freiwillig veröffentlicht werden, als auch wenn die Veröffentlichung ohne Einwilligung durch Dritte erfolgt. Als weiteres Problem wurden Widersprüche und Falschinformationen in den Datenbeständen aufgezeigt, während durch die strukturierte Form der Daten eine erhöhte Glaubwürdigkeit suggeriert wird. Informationen im Web aus Daten sind jedoch genauso kritisch zu betrachten wie im derzeitigen Web.

Abschließend ging diese Studienarbeit auf die Zukunftsaussichten von Linked Data ein. Als Hürde für das Wachstum an Daten im Web wurde das Problem „Database Hugging“ beschrieben. Hier muss ein Umdenken stattfinden, sodass nicht nur für Menschen aufbereitete Informationen, sondern auch Rohdaten veröffentlicht werden. Um strukturierte Daten zu generieren und zu verlinken, könnte man zukünftig die Methoden des sogenannten Social Webs mit Linked Data zu einem „Social Semantic Web“ kombinieren, sodass die Nutzer selbst Daten gemeinschaftlich erzeugen, so wie es im Social Web heute schon mit multimedialen Inhalten und deren Kategorisierung geschieht. Anwendungen und Tools mit Bezug zu Linked Data müssen ihren Kinderschuhen entwachsen und Web-Applikationen wie Wikis, Foren und Blogs müssen die Erzeugung von Linked Data implementieren, um eine Verbreitung bei der Masse der Web-Anwender

zu erreichen. Schritt für Schritt nähert sich das Web somit der Vision einer umfassenden, globalen Datenbank.

Glossar

Comma separated values

Dateiformat, bei dem Daten durch ein bestimmtes Zeichen getrennt in einer Textdatei gespeichert werden. 46

Domain Name System

Dienst zur Auflösung von Domainnamen zu IP-Adressen. 9, 46

Friend of a Friend (FOAF)

Ontologie zur Beschreibung von Freundschaftsbeziehungen. 46

Fully Qualified Domain Name

Vollständige Bezeichnung einer Domain. 9, 46

Hypertext Markup Language

Auszeichnungssprache zur Strukturierung von Dokumenten. 5, 46

Hypertext Transfer Protocol

Protokoll zur Übertragung von Hypertext-Dokumenten. 5, 46

Multipurpose Internet Mail Extensions

Kodierungsstandard zur Beschreibung von Nachrichteninhalten im Internet. 46

Notation 3

Von Tim Berners-Lee entwickelte Kurzschreibweise für das Resource Description Framework. 15, 46

Ontologie

Vokabular zur formalen Beschreibung von Wissen innerhalb eines Anwendungsbereichs 14

Open-Document-Format

Offener Standard für Office-Dokumente. 19, 46

Portable Document Format

Datenformat für den wiedergabegetreuen Austausch von Dokumenten. 46

RDF/XML

XML-Notation für das Resource Description Framework. 13, 15, 21, 23, 24

Resource Description Framework

Sprache zur formalen Beschreibung von Ressourcen im Web. 11, 12, 46

Resource Description Framework in attributes

Erweiterung von XHTML zur Einbettung von RDF-Tripeln mittels Attributen. 46

Resource Description Framework Schema

Ontologie-Beschreibungssprache. 21, 46

SPARQL Protocol and RDF Query Language

Abfragesprache für RDF. 46

Structured Query Language

Abfragesprache für relationale Datenbanken. 46

Uniform Resource Identifier

Eindeutiger Bezeichner für Ressourcen im Web. 8, 46

Web Ontology Language

Ontologie-Beschreibungssprache. 22, 46

World Wide Web

Eine Anwendung des Internets, aus weltweit verküpften und abrufbaren Hypertext-Dokumenten. 4, 47

World Wide Web Consortium

Von Tim Berners-Lee gegründetes Gremium zur Standardisierung von Web-Techniken. 7, 47

ZIP

Dateiformat zur Komprimierung von Dateien oder Verzeichnisbäumen. 10

Abkürzungsverzeichnis

| | |
|--------|--|
| CSV | Comma Separated Values 11, 36 |
| DNS | Domain Name System 9, 10 |
| FOAF | Friend of a Friend 19, 37, 41 |
| FQDN | Fully Qualified Domain Name 9 |
| HTML | Hypertext Markup Language 5, 6, 11, 18, 19, 24–26, 30 |
| HTTP | Hypertext Transfer Protocol 5, 9, 10, 23, 41 |
| MIME | Multipurpose Internet Mail Extensions 23 |
| N3 | Notation 3 15 |
| ODF | Open-Document-Format 19 |
| OWL | Web Ontology Language 7, 18, 22, 41 |
| PDF | Portable Document Format 35, 36 |
| RDF | Resource Description Framework 7, 8, 10, 12–16, 20, 22, 23, 25–28, 30–32, 40, 41 |
| RDFa | Resource Description Framework in attributes 12, 16 |
| RDFS | Resource Description Framework Schema 7, 18, 21, 22, 41 |
| SPARQL | SPARQL Protocol and RDF Query Language 7, 8, 10, 27, 28, 30, 40, 41 |
| SQL | Structured Query Language 28, 32 |
| URI | Uniform Resource Identifier 7–10, 15, 20, 22, 24–27, 29–32 |

W3C World Wide Web Consortium 7, 16, 22, 23, 28
WWW World Wide Web 4–6, 9–11, 23, 35, 40, 41

XHTML Extensible Hypertext Markup Language 12, 16
XML eXtensible Markup Language 12, 13, 15, 16, 36

Literaturverzeichnis

- [1] BERNERS-LEE, Tim: *Linked Data*. World wide web design issues. <http://www.w3.org/DesignIssues/LinkedData.html>. Version: Juli 2006. – Einsichtnahme: November 2009
- [2] 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. In: *Scientific American* 284 (2001), Nr. 5, S. 34–43
- [3] HAUSENBLAS, Michael: Exploiting Linked Data For Building Web Applications. In: *IEEE Internet Computing* 13 (2009), Nr. 4, 68–73. <http://sw-app.org/pub/exploit-lod-webapps-IEEEIC-preprint.pdf>
- [4] BERNERS-LEE, Tim ; FIELDING, Roy T. ; MASINTER, Larry: *Uniform Resource Identifier (URI): Generic Syntax*. Internet RFC 3986, Januar 2005
- [5] KUROSE, James F. ; ROSS, Keith W.: *Computer Networking: A Top-Down Approach*. 4. Addison Wesley, 2007
- [6] W3SCHOOLS: *RDF Tutorial*. <http://w3schools.com/rdf/default.asp>. – Einsichtnahme: November 2009
- [7] W3C: *N3 Einführung*. <http://www.w3.org/2000/10/swap/Primer>. – Einsichtnahme: November 2009
- [8] ADIDA, Ben ; BIRBECK, Mark ; MCCARRON, Shane ; PEMBERTON, Steven: *RDFa in XHTML: Syntax and Processing. A collection of attributes and processing rules for extending XHTML to support RDF. W3C Recommendation*. <http://www.w3.org/TR/rdfa-syntax/>, Oktober 2008
- [9] GRUBER, Tom: Ontology. In: LIU, Ling (Hrsg.) ; ÖZSU, M. T. (Hrsg.): *Encyclopedia of Database Systems*. Springer US, 2009, S. 1963–1965
- [10] DCMI USAGE BOARD: *DCMI Metadata Terms*. <http://dublincore.org/documents/dcmi-terms/>, Februar 2008
- [11] BRICKLEY, Dan ; MILLER, Libby: *The Friend Of A Friend (FOAF) Vocabulary Specification*. <http://xmlns.com/foaf/spec/>. Version: Dezember 2009

- [12] BRICKLEY, Dan ; GUHA, R.V. ; MCBRIDE, Brian: *RDF Vocabulary Description Language 1.0: RDF Schema*. W3C Recommendation. <http://www.w3.org/TR/rdf-schema>. Version: 2004
- [13] PATEL-SCHNEIDER, Peter ; HAYES, Patrick ; HORROCKS, Ian: *OWL Web Ontology Language Semantics and Abstract Syntax*. W3C Recommendation. <http://www.w3.org/TR/owl-semantics/>. Version: 2004
- [14] W3C: *Dereferencing HTTP URIs*. <http://www.w3.org/2001/tag/doc/httpRange-14/2007-05-31/HttpRange-14.html>. Version: 2007
- [15] BIZER, Chris ; CYGANIAK, Richard ; HEATH, Tom: *How to Publish Linked Data on the Web*. <http://sites.wiwiss.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/>. Version: 2007
- [16] SAUERMAN, Leo ; CYGANIAK, Richard ; VÖLKE, Max: *Cool URIs for the Semantic Web / DFKI GmbH*. Version: Februar 2007. <http://www.dfki.uni-kl.de/dfkidok/publications/TM/07/01/tm-07-01.pdf>. Deutsches Forschungszentrum für Künstliche Intelligenz GmbH, Februar 2007 (TM-07-01). – Technical Memo. – Written by 29.11.2006
- [17] HARTIG, Olaf ; BIZER, Chris ; FREYTAG, Johann-Christoph: *Executing SPARQL Queries over the Web of Linked Data*. In: *8th International Semantic Web Conference (ISWC2009)*, 2009
- [18] PRUD'HOMMEAUX, Eric ; SEABORNE, Andy: *SPARQL Query Language for RDF / World Wide Web Consortium*. Version: 2008. <http://www.w3.org/TR/rdf-sparql-query/>. 2008. – W3C Recommendation
- [19] BLUMAUER, Andreas (Hrsg.) ; PELLEGRINI, Tassilo (Hrsg.): *Social Semantic Web: Web 2.0 – Was nun?* Berlin : Springer, 2009

Abbildungsverzeichnis

| | | |
|-----|--|----|
| 2.1 | Schichtenmodell des Semantischen Webs. Quelle: World Wide Web Consortium (http://www.w3.org/2007/03/layerCake.png) | 7 |
| 3.1 | Einige Beispiels-URIs und deren Aufbau | 9 |
| 3.2 | Darstellung der Daten aus Tabelle 3.2 in einem RDF-Graphen | 13 |
| 3.3 | Der Daten-Browser Tabulator beim Anzeigen einer Ressource | 27 |

Tabellenverzeichnis

| | | |
|-----|--|----|
| 3.1 | Daten einer Beispiel-CSV-Datei. Aus der Datei geht die Bedeutung der Daten nicht hervor. | 11 |
| 3.2 | Beispiele für RDF-Tripel, angelehnt an natürliche Sprache. | 12 |
| 3.3 | Für RDFa neu in XHTML eingeführte Attribute und ihre Bedeutung. . | 17 |
| 3.4 | In XHTML enthaltene Attribute die für RDFa verwendet werden. . . . | 18 |
| 3.5 | Einige durch Dublin Core standardisierte Felder. | 19 |

Listings

| | | |
|------|---|----|
| 3.1 | Grundstruktur eines RDF/XML-Dokuments | 13 |
| 3.2 | Beschreibung eines Artikels in RDF/XML | 14 |
| 3.3 | Implizite Bestimmung des Typs durch Wahl des Eltern-Tags | 14 |
| 3.4 | Verlinkung auf eine andere Ressource | 15 |
| 3.5 | RDF-Tripel in Notation 3 | 16 |
| 3.6 | Ein mit RDFa um Metadaten angereicherter Online-Artikel | 16 |
| 3.7 | Hinterlegung von Metainformationen mit HTML | 18 |
| 3.8 | Probleme mit unterschiedlichen Werten im Attribut name | 18 |
| 3.9 | Beschreibung einer Person mittels FOAF | 19 |
| 3.10 | Definition einer Klasse mittels RDFS | 21 |
| 3.11 | Definition eines Prädikats mittels RDFS | 21 |
| 3.12 | Einfache SPARQL-Abfrage | 27 |