Semantisch homogene Beschreibung von Data-Warehouse-Metadaten mit RDF

Stefan Hartmann, Martin Weber

Wissenschaftliches Institut für Hochschulsoftware
der Universität Bamberg (ihb)
Feldkirchenstr. 21
96045 Bamberg
stefan.hartmann@ihb.uni-bamberg.de
martin.weber@stud.uni-bamberg.de

Abstract: Die Vernachlässigung einer unternehmensweit konsistenten Business-Intelligence-Strategie resultiert in multiplen Insellösungen bei Data-Warehouse-Systemen (DWH-Systemen). Eine direkte Vergleichbarkeit von Berichten aus verschiedenartigen DWH-Systemen ist im Allgemeinen aufgrund struktureller und semantischer Heterogenität nicht gegeben. Forschungsansätze widmen sich diesem Problem vor allem auf struktureller Ebene. Der in dieser Arbeit vorgeschlagene Ansatz betrachtet vorrangig semantische Aspekte bei der Beschreibung von DWH-Metadaten. Hierzu wird auf das im Semantic Web etablierte Resource Description Framework (RDF) zurückgegriffen. Es werden die Potenziale von RDF zur Beschreibung von DWH-Metadaten aufgedeckt und basierend auf diesen Ergebnissen ein RDF-Schema zur semantisch homogenen Beschreibung von DWH-Metadaten vorgeschlagen. Anhand einer beispielhaften DWH-Modellierung werden die Anwendbarkeit des RDF-Schemas sowie der erreichte Mehrwert hinsichtlich semantischer Homogenität aufgezeigt.

1 Motivation

Projekte zur Einführung eines Enterprise-Data-Warehouse konfrontieren Unternehmen nach wie vor mit großen organisatorischen und technischen Herausforderungen. Die horizontale und vertikale Differenzierung der unternehmerischen Gesamtaufgabe sowie die Autonomie der für eine Teilaufgabe verantwortlichen Stellen führen zu abteilungs- bzw. bereichsspezifischen Ausprägungen von Data-Warehouse-Systemen (DWH-Systemen). Neben heterogenen Datenschemata kennzeichnet vor allem ein verschiedenartiges Verständnis der Fachbegriffe die gewachsene DWH-Infrastruktur. Erweiterungen der Organisationsstruktur (bspw. durch Mergers&Acquisitions) erhöhen zusätzlich die Anzahl parallel betriebener DWH-Systeme [JW00, S.4ff.], [Wi04, S.317].

Ein dem DWH-Gedanken ursprünglich originäres Ziel, die horizontale sowie vertikale Datenintegration an einer zentralen Stelle, rückt hierdurch aus dem Fokus. Nur durch zeit- und ressourcenintensiven Abstimmungsaufwand können bei multiplen DWH-

Systemen aussagekräftige und vergleichbare Berichte generiert werden. In Folge dessen nimmt nicht nur die Aktualität, Qualität und Flexibilität der Datenanalyse ab, sondern auch der Wert einer abgeleiteten Information sinkt für den Anfrager [BM98, S.22ff.].

Ansätze, um Heterogenität auf DWH-Ebene zu begegnen, werden in der Literatur intensiv diskutiert. Einen wichtigen Ursprung finden sie in den Bestrebungen um das Thema *Schema-Management*, welches in den letzten Jahren unter dem Namen *Model-Management* wieder verstärkt Aufmerksamkeit findet [LN07, S.81, S.115ff.], [BM07]. Schema-Management konzentriert sich vor allem auf die Überwindung struktureller Heterogenität. Einen kombinierten Ansatz, um Daten aus mehreren DWH-Systemen zusammenzuführen, bietet das sog. *kollaborative Data-Warehousing*. Dieses gründet auf dem Prinzip der zusicherungsbasierten Integration (Correspondance Assertation [SPD92]) sowie dem Grundgedanken föderierter Datenbanken [Co97]. Durch Anwendung vorgegebener Integrationsregeln und Einbeziehung der Korrespondenz-Zusicherungen kann ein integriertes, virtuelles Schema über DWH-Systeme konstruiert werden [MWL07]. Der Ansatz der *verteilten Anfrage* (distributed query) kommt dagegen ohne eine virtuelle Zwischenschicht aus und bildet direkt ein komplexes SQL über die zu befragenden Datenquellen. Die verteilte Anfrage besitzt ihre Stärken bei der Anreicherung bestehender DWH-Systeme zur Anfragezeit mit externen oder nahezu Echtzeitdaten [Ec04, S.26]. Für ein homogenes Verständnis der Objekte in einem DWH wurde von LEHMANN und JASZEWSKI ein so genannter *Informationsnavigator* vorgeschlagen, der als zusätzliche Komponente zu einem DWH implementiert ist. Dieser fungiert als Hilfe- und Suchsystem über eine angeschlossene Lexikonkomponente, die ein konsolidiertes Begriffssystem beinhaltet [LJ99].

Semantische Aspekte bei der Überwindung der Heterogenität multipler DWH-Systeme wurden jedoch bisher nur peripher beleuchtet. In Anlehnung an die aktuellen Entwicklungen im Bereich des Semantic Web, die semantische Reichhaltigkeit von Dokumenten im Internet zu erhöhen, soll das im Semantic Web maßgebliche *Resource Description Framework* (RDF) auf seine Anwendbarkeit zur formalen und semantisch homogenen Beschreibung von DWH-Metadaten untersucht werden. In Kapitel 2 werden zunächst prinzipielle Aspekte der Metadaten im DWH untersucht und anschließend deren Korrelation mit der Mächtigkeit von RDF dargelegt (Kapitel 3). Den Aufbau eines prototypischen RDF-Schemas zur Beschreibung von DWH-Metadaten sowie dessen Anwendbarkeit offeriert Kapitel 4. Der Artikel schließt mit einem Fazit und Ausblick auf weitere Forschungsfragen (Kapitel 5).

2 Metadaten im DWH

Erst in Verbindung mit Metadaten gewinnen Daten an Bedeutung. Metadaten tragen insbesondere bei DWH-Systemen zu einem effizienten Betrieb und einer erfolgreichen Nutzung bei. Sie beschreiben den Inhalt und die Struktur eines DWH, sorgen für eine korrekte Interpretation und Verarbeitung der Daten und ermöglichen den Anwendern ein schnelles und sicheres Auffinden der benötigten Daten. Die einheitliche Beschreibung von Metadaten im DWH-Kontext stellt daher eine signifikante Herausforderung dar, um ein konsistentes Datenverständnis zu gewährleisten [KMU04, S.42f.].

Mit Standardisierungen für die Auszeichnung von Metadaten im DWH, deren prominentester Vertreter von der OMG das *Common Warehouse Metamodel*[1] (CWM) ist, wird versucht, eine homogene Metadatenbeschreibung zu erzielen. In den Standardisierungsvorschlägen werden semantische Aspekte jedoch häufig subaltern beachtet oder die vorgeschlagenen Standards werden aufgrund ihrer Komplexität nur zu einem geringen Teil in der Praxis berücksichtigt.

Notationen zum Entwurf von Softwaresystemen, wie z. B. die *Unified Modeling Language*[2] (UML), scheinen für eine formale und semantisch homogene Metadatenbeschreibung von DWH-Systemen ebenfalls ungeeignet. Die UML ist eine rein graphische Notation und unterstützt keine formale Definition [Je04].

AUTH identifiziert für DWH-Systeme acht Metadatenkategorien, die eine eingehende Metadatenbeschreibung erlauben [Au04, S.44ff.]. Für das hier zu entwickelnde Rahmenwerk einer semantisch homogenen Beschreibung von DWH-Metadaten, sollen vor allem die folgenden Kategorien Beachtung finden:

- Die Kategorie *Terminologie* beinhaltet Angaben zur Verwaltung von Fachbegriffen. Hierzu zählen Begriffsbenennung und -identität, Definitionen und Beziehungen zwischen Begriffen, Aufdeckung von Synonymen und Homonymen sowie Nennung des Verantwortlichen für einen Fachbegriff.

- Metadaten zur Benennung und Beschreibung von Datenstrukturen sind der Kategorie *Datenstruktur/-bedeutung* zugeordnet (Name und Beschreibung der Datenstruktur, Datentyp, Elementgröße etc.).

- Metadaten, welche der Verknüpfung von Systemkomponenten mit Organisationseinheiten dienen, werden unter der Kategorie *Organisationsbezug* subsumiert. Diese umfasst u. a. Data-Owner, Datenverwender, Berechtigungen.

- In der Kategorie *Datentransformation* werden Metadaten zusammengefasst, die Datentransformationsprozesse spezifizieren. Diese umfassen bspw. Angaben zu den Datenquellen und -zielen sowie zu den Transformationsschritten.

- Der Kategorie *Datenanalyse* gehören Metadaten zur Beschreibung der Analysemöglichkeiten an (Hypercube, Dimension, Kennzahl etc.).

Die Metadatenkategorien *Datenqualität*, *Metadatenhistorie* und *Systembezug* werden zur Wahrung der Übersichtlichkeit in diesem Artikel nicht näher untersucht.

[1] Der CWM-Standard ist unter http://www.omg.org/technology/cwm verfügbar.
[2] Der UML-Standard kann unter http://www.uml.org eingesehen werden.

3 Potenziale von RDF zur Metadaten-Beschreibung im DWH

Das vom W3C standardisierte *Resource Description Framework*[3] (RDF) stellt ein Metadaten-Rahmenwerk dar, um Ressourcen, insbesondere im World Wide Web (WWW), einheitlich zu annotieren sowie diese in einer maschinenlesbaren und -interpretierbaren Form zu speichern. Durch den Einsatz von RDF wird eine formale Semantik erreicht. Desgleichen werden Mechanismen bereitgestellt, um Informationen über die erfassten Ressourcen (Metadaten) miteinander in Beziehung zu setzen. Hierfür greift RDF auf etablierte Strukturen für den XML-basierten Datenaustausch zurück [Po03, S.1ff.], [LN07, S.295ff.].

Mit Hilfe von RDF soll ein Individuum Aussagen über beliebige Ressourcen treffen können. Eine RDF-Aussage besteht dabei aus einem Tripel, welches sich aus den drei Bestandteilen *Subjekt*, *Prädikat* und *Objekt* zusammensetzt. Das Subjekt ist allgemein gesprochen die Ressource, welche beschrieben werden soll. Diese ist durch einen *Uniform Resource Identifier*[4] (URI) genau bestimmt [Po03, S.21f.]. Das Prädikat beschreibt den Eigenschaftstyp der Ressource. Dies kann bspw. ein Attribut, eine Beziehung oder ein spezifisches Kennzeichen sein. Das Objekt entspricht dem Wert des Eigenschaftstyps des beschriebenen Subjekts und kann entweder durch ein Literal oder eine Ressource angegeben werden. RDF-Daten können verschiedenartig serialisiert werden. Zu den bekanntesten Notationen zählen N-Tripel, die eine Untermenge der N3-Notation bezeichnen, und RDF/XML. Als Standard-Repräsentation für RDF-Datenmodelle wurde vom W3C RDF-Graph festgesetzt [AH04, S.63ff.], [Po03, S.14ff.].

RDF offeriert lediglich ein fachlich neutrales Datenmodell. Es ist daher ein gemeinsames Schema erforderlich, um spezifische Vokabulare definieren zu können. RDF-Schema (RDFS) bietet ein domänen-neutrales Regelsystem, mit dem fachspezifische RDF/XML-Vokabulare erstellt werden können. Das Vokabular wird durch RDFS in einer typisierten Hierarchie organisiert (*Class* - *subClassOf*; *Property* - *subPropertyOf*) und ermöglicht somit die Bildung von Begriffshierarchien für die semantische Einordnung von Begriffen. Diese können sowohl auf die einzelnen Elemente der Metadatenformate als auch auf deren Inhalte angewendet werden. Ein RDF-Dokument kann anhand eines RDFS auf Validität geprüft werden [AH04, S.84ff.], [EE04, S.235ff.].

Nachfolgend soll aufgezeigt werden, wie die in Kapitel 2 vorgestellten Elemente der explizit genannten Metadatenkategorien auf die durch RDF und RDFS offerierten Potenziale zur Metadatenbeschreibung abgebildet werden können. Neben den vom RDF-Standard bereitgestellten Beschreibungskonstrukten wird dabei - soweit möglich - auf bestehende Beschreibungsfelder, wie sie bspw. durch *Dublin-Core*[5] (DC) angeboten werden, zurückgegriffen.

---

[3] Die W3C-Spezifikation des RDF ist unter http://www.w3.org/RDF abrufbar.
[4] Der URI-Standard kann unter http://www.ietf.org/rfc/rfc3986.txt eingesehen werden.
[5] Das Dublin-Core-Vokabular ist unter http://dublincore.org/documents/dcmi-terms beschrieben.

3.1 Metadatenkategorie Terminologie

Um eine Vereinheitlichung der Terminologie in betrieblichen Begriffssystemen zu erzielen, sind die verwendeten Fachbegriffe zu ermitteln und anschließend deren Benennung und Bedeutung zu normieren. Auf Instanzebene wird im RDF jeder Begriff mit einem Identifier (ID) bzw. URI gekennzeichnet. Die Begriffsbenennung kann anhand des RDFS-Elements *rdfs:label*, die Begriffsdefinition durch das DC-Beschreibungselement *dc:description* erfolgen. Selbiges gilt für die Definition der Klassen und Eigenschaftstypen im RDFS. Begriffsbeziehungen können mit Hilfe von im RDFS definierten Eigenschaftstypen *rdf:Property* sowie Sub-Klassenbeziehungen *rdfs:subClassOf* abgebildet werden.

Darüber hinaus sind Synonyme und Homonyme aufzudecken. Synonyme können anhand ihrer Beschreibung und einer gleichen Klassenzuordnung erkannt werden. Die Abbildung von Homonymen wird in RDF durch das Konzept der Begriffsidentität verhindert. Tabelle 1 bietet einen Überblick über die Elemente der Metadatenkategorie Terminologie und ihren Darstellungsmöglichkeiten in RDF und RDFS. Die im RDFS zu spezifizierenden Elemente (siehe Kapitel 4) sind kursiv dargestellt.

Metadatum	**Abbildung in RDF**	**Abbildung in RDFS**
Begriffsidentität	ID / URI	*ID / URI*
Begriffsbenennung	rdfs:label	rdfs:label
Begriffsdefinition	dc:description	dc:description
Begriffsbeziehungen		*rdf:Property*, *rdfs:subClassOf*
Synonyme	dc:description	gleiche Klassenzuordnung
Homonyme	keine Abbildung möglich	

Tabelle 1: RDF- und RDFS-Elemente in der Kategorie Terminologie

3.2 Metadatenkategorie Datenstruktur und -bedeutung

Datenstrukturen entstehen durch Aggregation einfacher Datenobjekttypen. Ein Datenobjekttyp ist eine Beschreibung eines Datenobjekts anhand seines Wertebereichs und seiner Klassifizierung. Ein Datenobjekt dient der informationstechnischen Speicherung eines Datenwerts. Der zugehörige Datenobjekttyp gibt Wertebereich und Klassifizierung des Datenobjekts an [FS06, S.307ff.], [Au04, S.49]. Name und beschreibender Text einer Datenstruktur werden über die DC-Elemente *dc:title*, *dc:description* bzw. über das RDF inhärente Beschreibungselement *rdf:description* dargestellt.

Die benötigten (Datenobjekt-)Typen werden durch Spezifikation entsprechender Klassen und Eigenschaftstypen eingeführt, welche selbst anhand von Beschreibungselementen (*rdfs:label*, *rdfs:comment*) semantisch charakterisiert sind. Dabei können durch die RDFS-Konstrukte *rdfs:range* und *rdfs:domain* Einschränkungen hinsichtlich der Kombinierbarkeit von Klassen und Eigenschaftstypen ausgedrückt werden. Für die

Angabe eines Datentyps wird im RDFS ein entsprechender Eigenschaftstyp bereitgestellt. Optional können Literale durch das RDF-Konstrukt *rdf:datatype* im RDF-Dokument weiter typisiert werden. Angaben zum Ersteller und Erstellungsdatum erfolgen durch die DC-Elemente *dc:publisher* und *dc:date*. Tabelle 2 fasst die Ausführungen dieser Metadatenkategorie zusammen.

Metadatum	Abbildung in RDF	Abbildung in RDFS
Name		dc:title
Beschreibung		dc:description
Typ		*rdfs:Class, rdf:Property* rdfs:label, rdfs:comment
Datentyp	rdf:datatype	*rdf:Property*
Ersteller		dc:publisher
Erstellungsdatum		dc:date

Tabelle 2: RDF- und RDFS-Elemente in der Kategorie Datenstruktur und -bedeutung

3.3 Metadatenkategorie Organisationsbezug

Der Kategorie Organisationsbezug gehören Metadaten an, die Auskunft über Entstehung und Verwendung der Daten im Rahmen der Geschäftsprozesse geben. Insbesondere das Metadatum Data Owner soll an dieser Stelle Beachtung finden. Dieses beschreibt, welche Organisationseinheit fachlich für die Dateninhalte verantwortlich ist. Im RDFS ist das Metadatum Data Owner durch einen zu definierenden Eigenschaftstyp (*rdf:Property*) sowie eine Klasse für die zu referenzierende Person (*rdfs:Class*) abzubilden. Die weiteren Elemente dieser Metadatenkategorie (z. B. Datennutzer, Berechtigungen) werden in diesem Artikel nicht beleuchtet.

3.4 Metadatenkategorie Datentransformation

Die Metadatenkategorie Datentransformation umfasst Metadaten, die den Weg der Daten aus dem Quellsystem durch die Ebenen eines DWH-Systems zu den Applikationen der Datenanalyse beschreiben. Exemplarisch soll aus dieser Kategorie das Metadatum Quelle in das zu formulierende RDFS eingebunden werden. Hierzu sind entsprechende Klassen und Eigenschaftstypen einzuführen, um Angaben über Quelldatenbanken (Datenbank, Tabelle, Tabellenspalte etc.) speichern zu können.

3.5 Metadatenkategorie Datenanalyse

Metadaten der Kategorie Datenanalyse sind vorwiegend auf die Unterstützung der Endanwender ausgerichtet und dienen der Steigerung von Effektivität und Effizienz bei der Datenanalyse. Hierzu zählen u. a. das schnelle Auffinden relevanter Objekte oder die Wiederverwendung bereits vorhandener Analysestrukturen und -verfahren. Für die Beschreibung der Metadaten zu Cube, Dimension, Dimensionshierarchiestufe (DHS)

sowie Kennzahl sind eigene Klassen im RDFS zu spezifizieren. Kennzahlbeziehungen können durch Eigenschaftstypen sowie Sub-Klassenbeziehungen wiedergegeben werden. Tabelle 3 gibt einen Überblick über die Metadaten dieser Kategorie und zeigt deren Abbildungsmöglichkeit im RDFS.

Metadatum	Abbildung in RDF	Abbildung in RDFS
Cube		*rdfs:Class*
Dimension		*rdfs:Class*
Dimensionshierarchiestufe		*rdfs:Class*
Kennzahl		*rdfs:Class*
Kennzahlenbeziehungen		*rdfs:subClassOf, rdf:Property*

Tabelle 3: RDF- und RDFS-Elemente in der Kategorie Datenanalyse

Es konnte aufgezeigt werden, dass ein Teil der erforderlichen Beschreibungskomponenten für DWH-Metadaten über RDF- bzw. DC-Standardelemente realisierbar ist. Die speziell für die Beschreibung von DWH-Metadaten im RDFS zu definierenden Klassen und Eigenschaftstypen (kursiv dargestellte Elemente) werden im folgenden Kapitel erarbeitet.

4 Ein RDF-Schema zur semantisch homogenen Metadaten-Beschreibung eines DWH

Die Beschreibung eines DWH erfolgt in multidimensionaler Form. Die zugehörige Metapher ist die eines Hypercubes (siehe z. B. [Si02]). Der Hypercube (kurz: Cube) bildet das Wurzelelement im nachstehend vorgeschlagenen RDF-Schema für DWH-Metadaten. Die Struktur sowie Terminologie der im RDFS spezifizierten Klassen und Eigenschaftstypen orientiert sich am Metamodell des semantischen Data-Warehouse-Modells (SDWM) [Bö01] und berücksichtigt die in Kapitel 3 vorgestellten Elemente zur DWH-Metadatenbeschreibung. Abbildung 1 zeigt die Gesamtsicht auf das RDF-Schema zur Beschreibung von DWH-Metadaten.

Zunächst werden für die in Kapitel 3.5 vorgestellten Elemente der Kategorie Datenanalyse die entsprechenden RDFS-Klassen und RDFS-Eigenschaftstypen definiert. Für eine lesbare Identifkation der Klassen und Eigenschaftstypen (Begriffsidentität, Kap. 3.1) wurden sprechende Namen als ID gewählt. Ein *Cube* wird durch *Dimensionen* aufgespannt, welche auf unterschiedlichen *Ebenen* angeordnete *DHS* beinhalten. DHS können durch *Dimensionsattribute* näher bestimmt werden. Die kursiv gedruckten Elementnamen sind in dem in Abbildung 1 visualisierten RDFS durch gleichnamige Klassen spezifiziert. Zur Sicherstellung eines semantisch korrekten Verständnisses einer Klasse besitzt jede Klasse eine genaue Benennung (*rdfs:label*) und Beschreibung (*rdfs:comment*). Beziehungen zwischen den angeführten Klassen können durch die Eigenschaftstypen *hatDimension*, *hatDimensionsHierarchieStufe*, *hatDimensionsAttribut*, *istEbene* und *verdichtetZu* ausgedrückt werden. Die Verwendungsmöglichkeit jedes Eigenschaftstyps (Property) ist dabei durch die Angabe

von Domäne (*rdfs:domain*) und Wertebereich (*rdfs:range*) determiniert. Die Domäneneinschränkung bestimmt, welche Klassen einen Eigenschaftstypen als Subjekt, die Bedingung für den Wertebereich, welche Klassen einen Eigenschaftstypen als Objekt besitzen dürfen. Somit ist gewährleistet, dass keine semantisch ungültigen Beziehungen bspw. zwischen Cube und Dimensionsattribut modelliert werden.

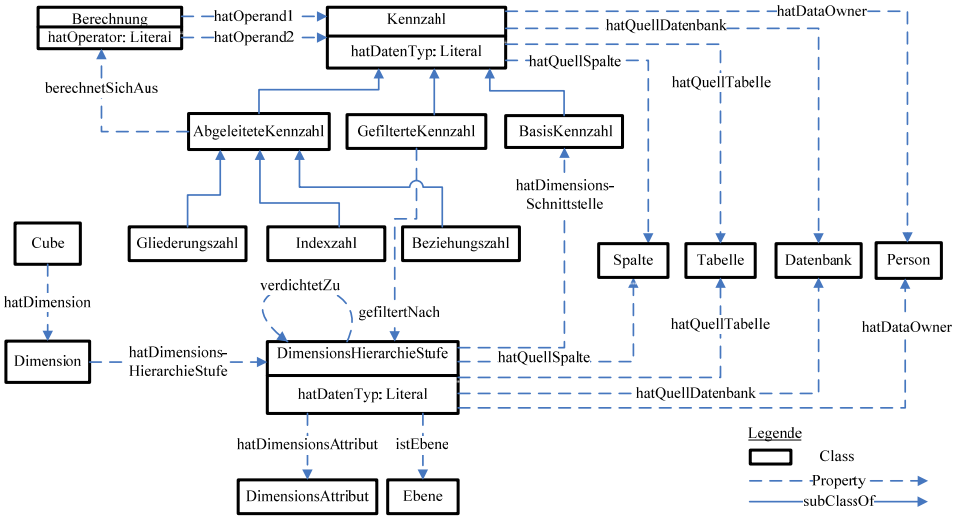

Abbildung 1: Gesamtsicht RDF-Schema

Im SDWM wird das in der Metadatenkategorie Datenanalyse eingeführte Element *Kennzahl* differenziert nach *Basiskennzahl*, *gefilterte Kennzahl* und *abgeleitete Kennzahl*. Letztere kann wiederum eine *Gliederungs-*, *Index-* oder *Beziehungszahl* sein. Im vorgeschlagenen RDFS für DWH-Metadaten sind entsprechende Klassen spezifiziert. Über Sub-Klassenbeziehungen sind die hierarchischen Abhängigkeiten zwischen den Kennzahltypen semantisch modelliert. Alle Kennzahlen erben die Spezifikation von Kennzahl. Eine abgeleitete Kennzahl (Bsp.: Gewinn = Umsatz - Kosten) ist durch den Eigenschaftstypen *berechnetSichAus* und die Klasse *Berechnung* bestimmt. Letzterer sind die Eigenschaftstypen *hatOperand1*, *hatOperand2* und *hatOperator* zugeordnet. Zulässige Operanden sind Kennzahlen (*rdfs:range*). Die Abgrenzung einer gefilterten Kennzahl (Bsp.: Umsatz der Filiale Süd) erfolgt anhand einer oder mehrerer DHS. Der Eigenschaftstyp *gefiltertNach* erlaubt diese Konstellation. Gemäß dem Metamodell des SDWM können DHS eine Dimensionsschnittstelle zu einer oder mehreren Basiskennzahlen aufweisen. Im RDFS ist dies durch den Eigenschaftstypen *hatDimensionsSchnittstelle* beachtet.

In der Metadatenkategorie Datentransformation werden Angaben zur (Daten-)Quelle gefordert (vgl. Kapitel 3.4). Hierzu sind im RDFS die Klassen *Datenbank*, *Tabelle* und *Spalte* spezifiziert. Über die zugehörigen Eigenschaftstypen *hatQuellDatenbank*, *hatQuellTabelle* und *hatQuellSpalte* können Beziehungen zu den genannten Klassen hergestellt werden. Die Domänen- und Wertebereichseinschränkungen erlauben dabei die Angabe von Quellsysteminformationen für Objekte der Klasse *Kennzahl* oder *DHS*.

Analog hierzu ist die Anforderung aus Kapitel 3.3 (Metadatenkategorie Organisationsbezug) abgebildet. Die Klasse *Person* dient der Angabe einer natürlichen Person, die Ansprechpartner für die Daten ist. Der Eigenschaftstyp *hatDataOwner* stellt eine Verknüpfung von den Klassen *Kennzahl* oder *DHS* zur Klasse Person her.

Die Gesamtstruktur des RDFS ist durch *dc:title*, *dc:description* sowie *dc:publisher* und *dc:date* gemäß den in Kapitel 3.2 geforderten Angaben charakterisiert. Die für DWH-Metadaten erforderlichen Datenobjekttypen wurden, wie beschrieben, in Form von Klassen und Eigenschaftstypen spezifiziert. Für die Angabe von Datentypen wurde der Eigenschaftstyp *hatDatenTyp* definiert. Nachstehend ist ein Ausschnitt des entwickelten RDFS in RDF/XML-Notation gezeigt:

```
<?xml version="1.0" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:dc="http://purl.org/dc/elements/1.1/">
   <rdf:Description
      rdf:about="http://www.semantic-metadata.com/dwhmd/elements/1.0/dwh-schema-simple#">
      <dc:title xml:lang="de">Data-Warehouse-Metadaten</dc:title>
      <dc:description xml:lang="de">Ein RDF-Schema fuer DWH-Metadaten.</dc:description>
      <dc:publisher>Martin Weber</dc:publisher>
      <dc:contributor>Stefan Hartmann</dc:contributor>
   </rdf:Description>
   <rdfs:Class rdf:ID="Cube">
      <rdfs:label>Cube</rdfs:label>
      <rdfs:comment>Diese Klasse dient der Beschreibung eines Cubes.</rdfs:comment>
   </rdfs:Class>
   <rdfs:Class rdf:ID="Dimension">
      <rdfs:label>Dimension</rdfs:label>
      <rdfs:comment>Diese Klasse definiert die Dimensionen eines Cubes.</rdfs:comment>
   </rdfs:Class>
   <rdf:Property rdf:ID="hatDimension">
      <rdfs:label>hatDimension</rdfs:label>
      <rdfs:comment>Verknüpfung eines Cubes mit einer Dimension.</rdfs:comment>
      <rdfs:domain rdf:resource="#Cube" />
      <rdfs:range rdf:resource="#Dimension" />
   </rdf:Property>
</rdf:RDF>
```

Zur Erläuterung der terminologischen Aspekte bei der Beschreibung von DWH-Metadaten (vgl. Kapitel 3.1) soll nachstehendes RDF-Beispiel dienen (Abbildung 2).

Oberhalb der Trennlinie in Abbildung 2 ist ein Ausschnitt des soeben eingeführten RDFS dargestellt, unterhalb eine Instanz des RDFS. Der Cube mit dem Namen *Verkaeufe* (Subjekt) verfügt über eine Dimension *Sortiment* (Objekt). Diese Aussage ist durch das Prädikat *hatDimension* komplettiert. Die Dimension *Sortiment* besitzt die DHS *Produkt*, *PG* (=Produktgruppe) und *Gesamt*, welche den Ebenen *1*, *2* und *3* zugewiesen sind. Die Prädikate *hatDimensionsHierarchieStufe* und *istEbene* verbinden die jeweiligen Subjekte und Objekte, sodass eine valide RDF-Aussage entsteht. Die DHS *Produkt* besitzt durch das Prädikat *hatDimensionsSchnittstelle* eine Verbindung zur Basiskennzahl *Verkaufsmenge*.

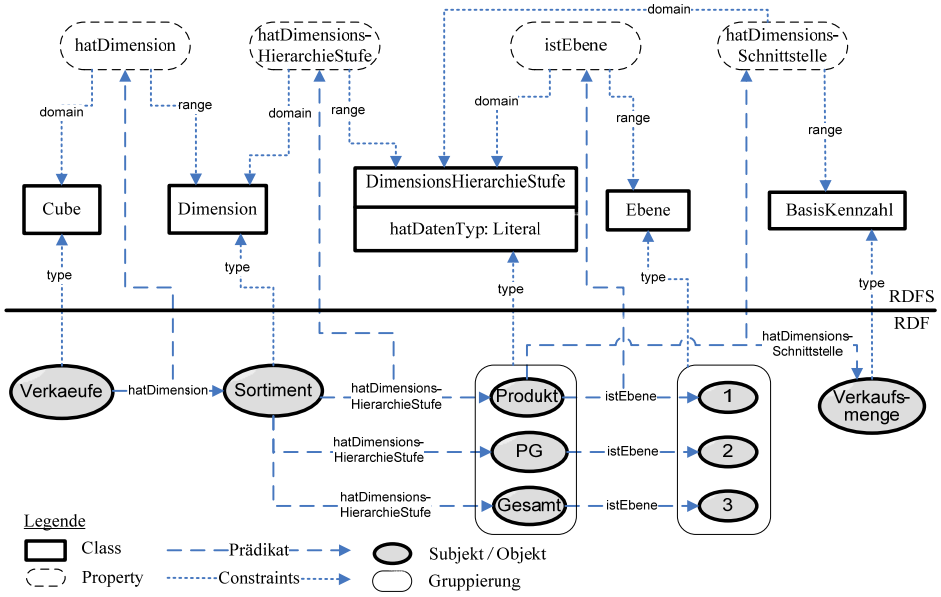

Abbildung 2: RDF- und RDFS-Ebene

Abbildung 2 weist außerdem die Domäne- und Wertebereichsbeschränkungen der im RDFS verankerten Eigenschaftstypen aus. Eine syntaktische Validitätskontrolle eines RDF oder RDFS ist mit dem vom W3C bereitgestellten RDF-Validator[6] möglich. Eine Prüfung gegen ein RDFS offeriert bspw. der RDF-Parser VRP[7]. Nachstehend ist das vorgestellte RDF-Beispiel in RDF/XML-Notation dargestellt:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dwhmd="http://www.semantic-metadata.com/dwhmd/elements/1.0/dwh-schema-simple#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://www.semantic-metadata.com/dwhmd/elements/1.0/dwh-schema-simple#">
  <dwhmd:Cube rdf:ID="Verkaeufe">
    <dc:description>Verkaufszahlen aller Filialen.</dc:description>
    <dwhmd:hatDimension rdf:resource="#Sortiment"/>
  </dwhmd:Cube>
  <dwhmd:Dimension rdf:ID="Sortiment">
    <dwhmd:hatDimensionsHierarchieStufe rdf:resource="#Sortiment_Produkt"/>
    <dwhmd:hatDimensionsHierarchieStufe rdf:resource="#Sortiment_Produktgruppe"/>
    <dwhmd:hatDimensionsHierarchieStufe rdf:resource="#Sortiment_Gesamt"/>
  </dwhmd:Dimension>
  <dwhmd:DimensionsHierarchieStufe rdf:ID="Sortiment_Produkt">
    <dwhmd:istEbene rdf:resource="#_1"/>
    <dwhmd:hatDimensionsSchnittstelle rdf:resource="#Verkaufsmenge"/>
```

[6] Siehe hierzu: http://www.w3.org/RDF/Validator.
[7] Siehe hierzu: http://athena.ics.forth.gr:9090/RDF/VRP.

```
  </dwhmd:DimensionsHierarchieStufe>
  <dwhmd:DimensionsHierarchieStufe rdf:ID="Sortiment_Produktgruppe">
    <dwhmd:istEbene rdf:resource="#_2"/>
  </dwhmd:DimensionsHierarchieStufe>
  <dwhmd:DimensionsHierarchieStufe rdf:ID="Sortiment_Gesamt">
    <dwhmd:istEbene rdf:resource="#_3"/>
  </dwhmd:DimensionsHierarchieStufe>
  <dwhmd:Ebene rdf:ID="_1"/>
  <dwhmd:Ebene rdf:ID="_2"/>
  <dwhmd:Ebene rdf:ID="_3"/>
  <dwhmd:Ebene rdf:ID="_4"/>
  <dwhmd:BasisKennzahl rdf:ID="Verkaufsmenge">
    <dwhmd:hatQuellDatenBank rdf:resource="#OpSYS1"/>
    <dwhmd:hatQuellTabelle rdf:resource="#Verkaufstabelle"/>
    <dwhmd:hatQuellSpalte>Menge</dwhmd:hatQuellSpalte>
    <dwhmd:hatDatenTyp rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">
            Integer</dwhmd:hatDatenTyp>
    <dwhmd:hatDataOwner>Hr. Datenverantwortlicher</dwhmd:hatDataOwner>
  </dwhmd:BasisKennzahl>
</rdf:RDF>
```

Die typisierte Hierarchie eines RDF-Schemas kann beliebig um Klassen und Eigenschaftstypen erweitert werden, um bspw. Elemente der in Kapitel 2 ausgeklammerten Metadatenkategorien zu ergänzen. Zudem ermöglicht das Namensraumkonzept die Einbindung weiterer Vokabulare in ein RDF [Po03, S.38ff.].

5 Fazit und Ausblick

Es wurde aufgezeigt, dass mit Hilfe des vorgestellten RDF-Schemas eine semantisch homogene Beschreibung von DWH-Metadaten möglich ist. Semantische Homogenität wird dabei zum einen durch die klar spezifizierte Bedeutung und Verwendungsmöglichkeiten der im RDFS definierten Klassen und Eigenschaftstypen unterstützt. Zum anderen fördert RDF eine semantisch einheitliche Annotation auf Instanzebene, da jedes Metadatum durch einen Identifier bestimmt ist, über Beschreibungsfelder genau erläutert ist und über Eigenschaftstypen klar definierte Verknüpfungen zu anderen Metadaten herausgestellt sind. Auch sprachliche Defekte wie Synonyme und Homonyme können aufgedeckt werden. Nachteilig zeigt sich unter anderem, dass in RDF sog. Constraints nur bedingt spezifiziert werden können. Eine Angabe von Kardinalitäten ist bspw. nicht vorgesehen. Auch eine Unterstützung von Inferenzregeln findet sich in RDF nicht. Diese Konstrukte sind mächtigeren Ontologiesprachen wie der *Web Ontology Language*[8] (OWL) oder DAML+OIL[9] vorbehalten. RDF verfügt jedoch über eine ausreichende Mächtigkeit, um als Basis für höherwertige Ontologiesprachen zu dienen. Das beschriebene RDFS kann folglich als Grundstock für die Beschreibung von DWH-Metadaten mit semantisch reichhaltigeren Konzepten fungieren.

[8] Detaillierte Informationen zur OWL finden sich unter: http://www.w3.org/TR/owl-features.
[9] Eine Beschreibung der DAML+OIL steht unter http://www.daml.org/2001/03/daml+oil-index.html bereit.

Literaturverzeichnis

[AH04] Antoniou, G.; van Harmelen, F.: A Semantic Web Primer. MIT Press, Cambridge 2004.

[Au04] Auth, G.: Prozessorientierte Organisation des Metadatenmanagements für Data-Warehouse-Systeme. Books on Demand GmbH, Nordersted 2004.

[BM07] Bernstein, P. A.; Melnik, S.: Model Management 2.0: Manipulating Richer Mappings. In: Proceedings of the 2007 ACM SIGMOD international conference on Management of data. Beijing 2007, S.1-12.

[BM98] Behme, W.; Mucksch, H.: Die Notwendigkeit einer entscheidungsorientierten Informationsversorgung. In: Mucksch, H.; Behme, W.: Das Data Warehouse-Konzept: Architektur-Datenmodelle-Anwendungen, 3. Auflage, Gabler, Wiesbaden 1998, S.3-31.

[Bö01] Böhnlein, M.: Konstruktion semantischer Data-Warehouse-Schemata. Deutscher Universitätsverlag, Wiesbaden 2001.

[Co97] Conrad, S.: Föderierte Datenbanksysteme - Konzepte der Datenintegration. Springer, Berlin 1997.

[EE04] Eckstein, R., Eckstein, S.: XML und Datenmodellierung - XML-Schema und RDF zur Modellierung von Daten und Metadaten einsetzen. dpunkt, Heidelberg 2004.

[Ec04] Eckerson, W.: In Search of a Single Version of Truth: Strategies for Consolidating Analytic Silos. TDWI Report Series, August 2004. http://download.101com.com/pub/TDWI/Files/TDWI_DMC_Report.pdf (Abruf am 23.11.2007).

[FS06] Ferstl, O. K.; Sinz, E. J.: Grundlagen der Wirtschaftsinformatik. 5. Auflage, Oldenbourg, München 2006.

[Je04] Jeckle, M. et al.: UML 2.0: Evolution oder Degeneration? In: ObjektSpektrum 03/2004, S.12-19.

[JW00] Jung, R.; Winter, R.: Data Warehousing: Nutzungsaspekte, Referenzarchitektur und Vorgehensmodell. In: Jung, R.; Winter, R. (Hrsg.) Data Warehousing Strategie - Erfahrungen, Methoden, Visionen. Springer, Heidelberg 2000, S.3-20.

[KMU04] Kemper, H.-G.; Mehanna W.; Unger, C.: Business Intelligence - Grundlagen und praktische Anwendungen - Eine Einführung in die IT-basierte Managementunterstützung. Vieweg, Wiesbaden 2004.

[LJ99] Lehmann, P.; Jaszewski, J.: Business Terms as a Critical Success Factor for Data Warehousing. In: Gatziu, S.; Jeusfeld, M.; Staudt, M.; Vassiliou, Y. (Hrsg.): Proceedings Workshop on Design and Management of Data Warehouses, Heidelberg 1999, S.7.1-7.5.

[LN07] Leser, U.; Naumann, F.: Informationsintegration - Architekturen und Methoden zur Integration verteilter und heterogener Datenquellen. dpunkt, Heidelberg 2007.

[MWL07] Matheis, T.; Werth, D.; Loos, P.: Kollaboratives Data Warehousing - Konzeption und prototypische Realisierung flexibler Schema- und Datenintegration. In: Oberweis, A.; Weinhardt, C. et al. (Hrsg.): eOrganisation - Service-, Prozess-, Market-Engineering: 8. Internationale Tagung Wirtschaftsinformatik - Band 1. Universitätsverlag Karlsruhe, Karlsruhe 2007, S.569-586.

[Po03] Powers, S.: Practical RDF - Solving Problems with the Resource Description Framework. O'Reilly, Beijing 2003.

[Si02] Sinz, E. J.: Data Warehouse. In: Küpper, H.-U.; Wagenhofer, A.: Handwörterbuch Unternehmensrechnung und Controlling. 4. Auflage, Schäffer-Poeschel, Stuttgart 2002, S.309-318.

[SPD92] Spaccapietra, S.; Parent, C.; Dupont, Y.: Model Independent Assertions for Integration of Heterogeneous Schemas. In: The VLDB Journal - The International Journal on Very Large Data Bases 1 (1992) 1, S.81-126.

[Wi04] Winter, R.: Architektur braucht Management. In: Wirtschaftsinformatik 46 (2004) 4, S.317-319.