Czech Technical University in Prague

Doctoral Thesis Statement

Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Computer Science and Engineering

Transforming Relational Data into Ontology Based RDF Data

by

Martin Švihla

A doctoral thesis submitted to
the Faculty of Electrical Engineering, Czech Technical University in Prague,
in partial fulfilment of the requirements for the degree of Doctor.

Ph.D. Programme: Electrical Engineering and Information Technology
Branch of study: Computer Science and Engineering

June 2007

This doctoral thesis was carried out as a part of a full-time Ph.D. study at the Department of Computer Science and Engineering of the Faculty of Electrical Engineering of the Czech Technical University in Prague.

Candidate: Ing. Martin Švihla
Department of Computer Science and Engineering
Faculty of Electrical Engineering
Czech Technical University in Prague
Karlovo náměstí 13
121 35 Prague 2, Czech Republic

Supervisor: Ivan Jelínek
Department of Computer Science and Engineering
Faculty of Electrical Engineering
Czech Technical University in Prague
Karlovo náměstí 13
121 35 Prague 2, Czech Republic

Reviewers:

....

....

....

The doctoral thesis statement was distributed on ....

The defence of the doctoral thesis will be held on .... at .... before the Board for the Defence of the Doctoral Thesis in the branch of study Computer Science and Engineering in the meeting room No. .... of the Faculty of Electrical Engineering of the Czech Technical University in Prague.

Those interested may get acquainted with the doctoral thesis concerned at the Dean Office of the Faculty of Electrical Engineering of the Czech Technical University in Prague, at the Department for Science and Research, Technická 2, Prague 6.

Prof. Ing. Pavel Tvrdík, CSc.

Chairman of the Board for the Defence of the Doctoral Thesis
in the branch of study Computer Science and Engineering

Contents

1 Introduction: Motivation and goals

The World Wide Web (WWW) is the largest knowledge database that mankind has ever created. Its simple and open principles enable people to publish or access information very easily. On the other hand, as the web continuously grows, we face problems with the amount of information and its organisation. It is becoming more and more difficult to find relevant information, manage heterogeneous knowledge sources, and transparently aggregate several information providers.

There are several initiatives to improve the situation. One of them is the semantic web [6], which would give more structure and computer-understandable meaning to the data on the WWW. The semantic web is not a separate web but an extension of the current one, in which information is given a well-defined meaning, better enabling computers and people to work in cooperation [6].

Realization of the semantic web is still in the very early stages and widespread use is yet to be achieved. Its success also depends on mass creation of semantic metadata that is expected to cover the existing web by machine-understandable meaning. To achieve this goal, several approaches for manual and/or automatic annotation of existing web resources were designed (see Section 3).

This is the focus of our work: generation of semantic web metadata. When investigating the nature of the research field, we identified the following points:

- A large part of the web content is generated from relational data stored in RDBMSes. It is not possible to give an exact ratio of dynamic to static web pages, but some sources (e.g. [13]) claim that dynamic web pages outnumber static ones 100 to 1.
- The semantic web standards (RDF – Resource Description Framework [2], RDF-Schema (RDFS) [8], and OWL – Web Ontology Language [23]) are already deployed.
- The semantic web technology is considered complex and complicated by people outside the semantic web community.

To address these issues, in this work we propose a system for generating metadata directly from relational database (RDB). Our *novel data transformation model* is based on the mapping of a relational database schema to an ontology. This transformation adds explicit semantics to relational data by means of semantic web technologies.

Since our research is focused on the semantic web, our proposal addresses particular *semantic web technologies*. The format of produced metadata is RDF

(Resource Description Framework [2]) and the format of ontology language is RDFS (RDF-Schema [8]). We also aim to support OWL [23].

Regarding the complexity of semantic web technologies, we put forth a strong effort to design a data transformation model that is *usable* even for web developers without a deep knowledge of RDF and RDFS. It is suggested [14] that the creation of metadata should be as simple as the creation of HTML. Unfortunately, this is not yet the case. The success of the web is based on its simplicity, but semantic web technologies are much more complex. One mission in semantic web research today is to hide this complexity from the web community [18].

Another aspect we observe is *performance* of the proposed solution. The architecture and algorithms of our model are designed to be efficient from this point of view – our goal is to make RDF production time-efficient.

In our work we formally designed a model for such a data transformation and proposed two XML languages based on the model. Then we implemented a data transformation prototype, which was later deployed in several experimental case studies. We also performed an extensive performance analysis, which proved the high performance of our approach and brought a completely new view to the field of RDB to RDF data transformation and RDF storage.

There are many possible uses of such a data transformation. In [26] there are mentioned, for example: scientific databases, syndication, and community web portals. We can add to this list e-commerce catalogues, digital libraries, and other types of web applications, but the general idea is clear: the relevant target group for such transformation is a large number of dynamic web sites built on top of RDBs.

The steps we took to reach our goals are briefly described in Section 2, while Section 3 introduces related work in the field, and thesis contributions are summarised in Section 4.

2 Data transformation model

2.1 Design rationale

As discussed in Section 1, the goal of this work is to enable *a transformation of relational data into an RDF document.* The transformation is based on *a mapping that maps a relational database schema into an RDFS ontology.*

Generally we can identify the two distinct problems (Figure 2.1) that we will address as: *(i) schema mapping* and *(ii) instance transformation.*

Schema mapping takes two schemas from the same knowledge domain as input and produces a mapping between elements of the schemas that correspond "semantically" to each other. *Instance transformation* describes queries that transform values from a data source into a different target data in accordance with the previously defined mapping between schemas.

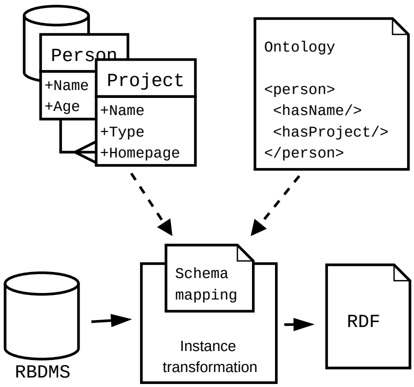

Figure 2.1: Data transformation schema

Following these two steps, we divided our data transformation model architecture into two separate parts: the *schema mapping layer* and the *template layer* (Figure 2.2).

In the schema mapping layer, which corresponds to the schema mapping process, a relational schema is mapped into an RDFS ontology. This is done by *mapping elements* that connect relating parts of a database schema and ontology. A set of mapping elements creates a *schema mapping document.*

The template layer corresponds with the instance transformation process. On this level, references to mapping elements from the mapping layer are combined to create a *template document*, which is a query for a transformation of relational data into an RDF document.

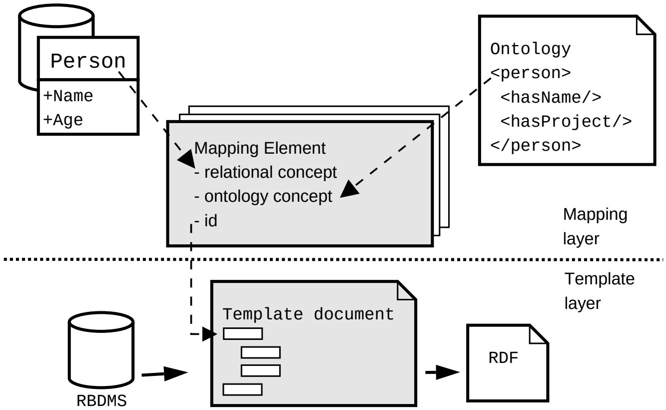

Figure 2.2: Data transformation architecture

Formal issues of these two layers and their cooperation are detailed in the thesis.

2.2 Data transformation languages

To enable the schema mapping and data transformation, we developed two XML languages. One is for the schema mapping documents and the other is for template documents.

Schema mapping language is used to describe the mapping between a database schema and given ontology. It can describe elements, which map concepts of a database schema to classes and properties of an ontology.

The schema mapping language is designed to: *(i)* describe mapping elements between a database schema and structure of a given ontology, *(ii)* enable control over these elements, *(iii)* store information for making a complete RDF document, and *(iv)* establish connection with a specified relational database.

Listing 2.1 gives a very simple example of this language. It maps the ontology class `Person` to a corresponding relational concept by SQL query and the ontology property `surname` to the database column `family_name`.

Listing 2.1: Datatype mapping property example

```
<Class templateName="person" rdfLabel="Person" sql="SELECT * FROM
    person p, department d WHERE d.id = p.id_department">
  <Property templateName="surname" rdfLabel="surname" sqlName="
      family_name" />
</Class>
```

The *template language* is to serialise a template document. A template document forms a query for the transformation of relational data to RDF. A document is a tree consisting of template elements, which are references to mapping elements from a particular mapping document. Referenced mapping elements translate the query to relational database queries and data from returned relations are composed according to the template document structure. This means that the template documents form a query but also specify a structure of resulting data.

The example in Listing 2.2 refers to the previous mapping fragment. It produces an RDF document and writes all persons from a table `PERSON` to it. All produced RDF instances have the type `Person` and contain the RDF property `surname` with a corresponding value from the database.

Listing 2.2: Template property example

```
<putInstance name="person">
   <putProperty name="surname" />
</putInstance>
```

2.3 Implemented prototype and supporting tools

We implemented a set of tools to enable the described data transformation: the data transformation processor *METAmorphoses*, the web publishing tool *RDF-Shout*, and the *schema mapping editor*. All applications are written in Java to achieve platform independence.

METAmorphoses transforms data from a relational database into RDF documents according to a mapping. To enable this, *METAmorphoses* processes schema mapping and template documents written in our XML languages. The processor employs a very efficient algorithm based on our data transformation model. It is designed as a stream processor without the use of any extensive RDF programming library so that it has *higher performance* than similar solutions in the field. We also addressed *usability* perspective in the implementation. The tool is designed to hide the complexity of the semantic web technologies into the schema mapping layer, while exposing the simple template layer to the programmer. This makes *METAmorphoses* easy to use and easily adaptable by current web development process (as described in detail in the thesis).

RDF-Shout is a simple web application designed to publish data from a relational database as RDF documents on the web. It uses *METAmorphoses* processor library as a core of the data transformation. The application is written using Java Servlet technology and requires an application server with a servlet container (e.g. Apache Tomcat [12]).

Schema mapping editor is a GUI application meant to simplify the schema mapping process. It helps the human user create mapping documents. The user interface of the application is depicted in Figure 2.3. The editor connects a specified database and provides its structure in the left window. An ontology can be viewed in the right window. The whole application is designed to use the *drag'n'drop* concept. Schema concepts from both sides can be dragged to the centre, where a visual representation of a mapping document is built and the editor asks for values of mandatory fields. This mapping document can be saved in XML format and used directly by the *METAmorphoses* processor.

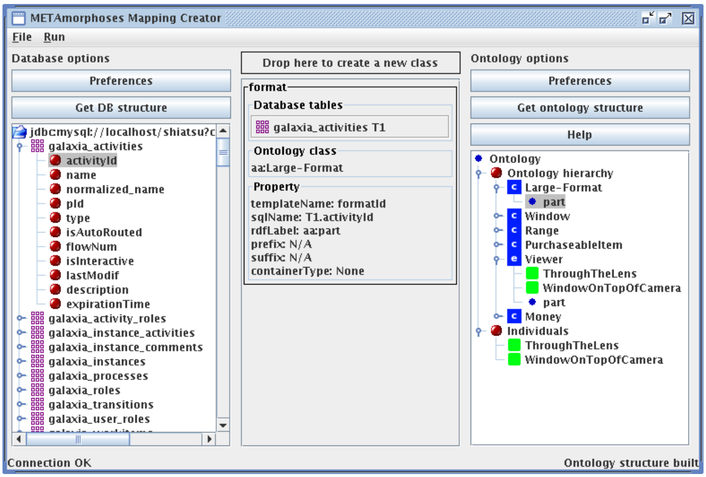

Figure 2.3: Schema mapping editor

2.4 Performance analysis

We executed a set of performance tests to compare our transformation processor with similar applications built on other design principles. The tests showed how the design affects performance. We compared five different systems in our experiments: three tools for the RDB to RDF transformation (*METAmorphoses*, D2RQ

[22], and SquirrelRDF [24]) and two native RDF repositories with RDB back-end (Jena [10] and Sesame1 [9]). Moreover, we performed two different tasks with D2RQ and Jena in the most of tests by querying dataset with both SPARQL and graph API. To compare these tools we used micro-benchmarks on an extensive dataset (DBLP computer science bibliography [17]).

METAmorphoses had the best performance in most tests (12 out of 13) and showed dominance in all of the tested performance aspects. This proves that our concept of data transformation has a higher performance than other data transformation solutions and than native RDF repositories. This means that on-the-fly data transformation based on our ideas can be accomplished faster than queries over native RDF repository. Thus, it is unnecessary to migrate relational data into RDF repositories in order to publish them later as RDF.

2.5 Case studies

We also conducted various experiments with the implemented system in the process of completing this thesis. These experiments (detailed in the thesis) were designed to prove our ideas in a practical environment and to test implementation and usability issues. Some of the experiments became independent projects based on the METAmophoses.

To give a brief overview to these case studies, we *(i)* deployed the *METAmorphoses* to provide RDF metadata about the Department of Computer Science and Engineering at CTU in Prague. Then we used these metadata to *(ii)* experiment with semantic information retrieval. We built the search engine, which crawls through HTML pages and digs for linked RDF resources. The crawler downloads RDF, indexes it, and stores it in the RDF knowledge base. RDF statements are indexed by a URL of the web page to which they belong, thus there is a link between RDF data and web pages in the knowledge base. The end-user can query this knowledge base by means of the simple web interface with the semantic search capability. In the last experiment we *(iii)* implemented an information portal capable of aggregating and publishing RDF metadata about academic publications.

These case studies were supported by grants and their results have also been published.

3 Related work

The topic of our work is closely related to many research areas. Here we mention only approaches directly relating to our work; more extensive descriptions of the related work can be found in the thesis.

There are several studies of relational data to RDF migration. These approaches differs in purpose, architecture, usability, algorithm efficiency, and so on. In this section, we examine major contributions to this field and their main attributes.

The problem of exposing relational data in RDF is usually solved through two different approaches:

- **A relational database is queried by an RDF query language.** Mapping is created between relational database schema and predefined ontology and based on this mapping, relational data can be queried by semantic queries. A result of the query is a set of RDF statements.
- **A relational database content is transformed into RDF.** A subset of relational database content is transformed into RDF. Resulting RDF can be stored in static RDF documents or in a native RDF database.

Both approaches are discussed in the following subsections along with early proposals in the field. We must stress that our approach does not fit into this simple classification because it exists between these two main groups. We do not use any standardised semantic query languages to fetch data from relational databases nor do we transform entire database content into RDF. Instead we provide the means of transforming any part of a database into RDF using our own query mechanism.

3.1 Early proposals

The semantic web is sometimes proposed as a huge distributed database. In 1998 (one year after the first RDF proposal), Tim Berners-Lee proposed the first work on exposing a relational database on the semantic web [4]. This document compares the relational model with the RDF data model and also discusses mapping issues. However, this mapping of relational data to RDF is rather basic and completely omits, for example, a concrete schema representation.

A more complex proposal is found in [3], in which the authors describe a naïve approach for mapping RDBMS data onto RDF data. The proposal covers concept naming, relationships, datatypes and other problems. However, the work is still mainly on the level of the RDF data model and not on the level of schema mapping.

3.2 RDF gateways to relational databases

SquirrelRDF [24] is a tool which allows relational databases to be queried using SPARQL [20]. It is simply an implementation of *the naïve RDB to RDF mapping*, described in [3], thus an ontology is not considered. When mapping is created, RDB can be queried by SPARQL [20]. A result of the SPARQL query over RDB can be RDF (if a *CONSTRUCT* query is used). The SquirrelRDF uses Jena [10] API to perform SPARQL queries.

The work proposed in [16] extends the naïve approach, focusing on linking relational and RDFS schemata. The system FDR2 is designed for integration of relational-like information resources with RDFS-aware systems by linking a relational database schema with a predefined domain ontology. This approach describes a relational schema (all of its concepts and relationships) by RDFS and uses this description to join relational schema and given ontology. The system requires an RDFS reasoner to process the mapping, which poses a serious performance problem. Moreover, the work does not solve the problem of data transformation – it is designed to allow querying a relational database by concepts from a predefined domain ontology.

A very similar approach is DartGrid, proposed in [11], which should provide a web-scale integration infrastructure for distributed relational database resources. The integration platform uses RDF/OWL to define mediated schemas, thus it is necessary to transform relational data into RDF according to existing OWL ontology. In this process, first the database semantic is described by RDF/OWL using a set of predefined rules. This source data semantic is later mapped into a shared ontology to allow semantic querying of the database. For this querying, the authors developed their own query language called Q3. The mapping approach has the same structure as the previous one, although it provides more sophisticated mapping according to the use of OWL, and it suffers from the same drawbacks.

Another contribution in this group is Federate [19], which is designed to provide a consistent RDF interface to relational databases. The approach uses RDF agents with the ability to query a relational database with an application-specific schema. First the very simple mapping is created between relational database schema and ontology concepts and then a database is queried by an RDF query language. The query is translated to SQL in accordance with the mapping. However, returned data take the form of a relation.

A very recent approach to DB to RDF migration is R2DQ [22]. This work introduces D2R, a declarative language used to describe mappings between relational database schemas and OWL ontologies. The mapping process consists of four steps: *(i)* selection of a record set from the database using SQL, *(ii)* grouping of the record sets, *(iii)* creation of class instances and identifier construction and *(iv)* mapping of the grouped record set data to instance properties. The mapping is used by a processor that transforms relational data into RDF, emitting RDF/XML [2], N3 [5] or N-Triples [1]. The processor is based on JENA API [10]. In the last version (0.5) a database can be queried by SPARQL.

3.3 Relational database to RDF transformers

The work described in [25] introduces a system called KAON REVERSE, which is intended to be a plug-in to the large semantic web framework KAON [7]. Using this system, data from a relational database are transformed into a RDF knowledge base based on the mapping of a logical database schema into an existing ontology. This approach begins by transforming the relational database model into a corresponding ontological structure based on F-Logic, which is then used for mapping the content of the database into an RDF knowledge base. This knowledge base can be queried by RDF query languages and resulting RDF statements can be published on the web. When mapping is created from this approach, the final layout of produced RDF is also defined.

Another work on relational database to RDF transformation is a part of the project MuseumFinnland [15]. This approach focuses mainly on interoperability between different database maintainers. In [21] a two-phase data transformation is described. A database content is transformed into an XML repository conforming to an XML Schema. These XML data are then transformed into RDF metadata, which are semantically validated against a set of predefined RDFS ontologies. The entire relational database is migrated into RDF without any possibility of transformation control. This approach uses a conceptual database schema for mapping.

D2RQ [22], mentioned in the previous subsection, also belongs in this category. In its early version (2004), it was just a system exporting data from a relational database into RDF.

We can summarise that all approaches discussed above use some RDF API to handle RDF data and ontologies and that the layout of produced RDF is rather static.

4 Summary and contributions

This thesis presents a new model for data transformation from a relational database to an ontology-based RDF metadata. Such transformation allows one to publish relational data without explicit semantics as RDF data but with explicit semantics defined by an ontology. The model divides the transformation process into two phases: schema mapping and instance transformation.

The main theoretical contributions of the thesis include:

- The identification of correspondences between the relational model and RDFS and also between the structure of relational data and RDF.
- The design of a formal model architecture, which is divided into two layers: one for schema mapping and the other for instance transformation based on the schema mapping.
- The proposal of two XML languages based on the formal model. *Schema mapping language* is capable of describing schema mapping between relational schema and RDFS, while *template language* is designed to create RDF documents from a relational database.
- The proposal of a high performance algorithm for the data transformation.

We verified our theoretical approach by conducting a formal examination of the model capability. We showed that a result of our data transformation is an RDF document with a connection to the used ontology. We also explored relational completeness of our approach and its compatibility with RDF and RDFS standards. We showed that our model supports all features of used formalisms and data models.

The practical outcome of this work is the implementation of a data transformation processor and supporting tools that enable web developers to enrich their RDB-backed websites with RDF metadata. This system was deployed in various case studies to prove our concepts and the performance analysis allowed comparison of our system with other approaches.

Another very important contribution is a conclusion of the performance analysis. We showed that our approach is faster not only than similar data transformation tools but also than native RDF repositories. This means that an on-the-fly data transformation based on our concepts can be executed faster than queries over current native RDF repositories. Proving this, we can conclude that it is unnecessary to migrate relational data to RDF repositories in order to publish them later as RDF.

Bibliography

[1] D. Beckett. RDF Test Cases. W3C Recommendation. Available at: http://www.w3.org/TR/rdf-testcases/#ntriples, February 2004.

[2] D. Beckett. RDF/XML Syntax Specification (Revised). http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/, February 2004.

[3] D. Beckett and J. Grant. Semantic Web Scalability and Storage: Mapping Semantic Web Data with RDBMSes. SWAD-Europe deliverable, http://www.w3.org/2001/sw/Europe/reports/scalable_rdbms_mapping_report, February 2003.

[4] T. Berners-Lee. Relational Databases on the Semantic Web. http://www.w3.org/DesignIssues/RDB-RDF.html, September 1998.

[5] T. Berners-Lee. Notation 3. http://www.w3.org/DesignIssues/Notation3, March 2006.

[6] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5), May 2001.

[7] E. Bozsak et al. KAON - Towards a Large Scale Semantic web. In *Third International Conference E-Commerce and Web Technologies, EC-Web 2002*, France, 2002.

[8] D. Brickley. RDF Vocabulary Description Language 1.0: RDF Schema. http://www.w3.org/TR/rdf-schema/, February 2004.

[9] J. Broekstra, A. Kampman, and F. van Harmelen. Sesame: A generic architecture for storing and querying rdf and rdf schema. In *Proceedings of the First International Semantic Web Conference*, page 54Â68. Springer, 2002.

[10] J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson. Jena: implementing the semantic web recommendations. In *WWW Alt. '04: Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 74–83, New York, NY, USA, 2004. ACM Press.

[11] H. Chen, Z. Wu, G. Zheng, and Y. Mao. Rdf-based schema mediation for database grid. In *GRID '04: Proceedings of the Fifth IEEE/ACM International Workshop on Grid Computing (GRID'04)*, pages 456–460, Washington, DC, USA, 2004. IEEE Computer Society.

[12] J. Goodwill. *Apache Jakarta-Tomcat.* Apress, Berkely, CA, USA, 2002.

[13] S. Handschuh, S. Staab, and R. Volz. On Deep Annotation. In *12th International World Wide Web Conference, WWW 2003*, 2003.

[14] J. Hjelm. *Creating the Semantic Web with RDF.* Wiley Computer Publishing, New York, 2001.

[15] E. Hyvonen et al. Finnish Museums on the Semantic Web: The user's Perspective on MuseumFinland. In *Museums and the Web Conference 2004*, Arlington, VA, USA, 2002.

[16] M. Korotkiy and J. L. Top. From relational data to rdfs models. In *International Conference on Web Engineering 2004*, Munich, 2004.

[17] M. Ley. DBLP Bibliography. http://www.informatik.uni-trier.de/ ley/db/, May 2003.

[18] M. Lytras. An interview with Eric Miller. *SIGSEMIS: Semantic Web and Information Systems*, 1(2), July 2004.

[19] E. Prud'hommeaux. Optimal RDF access to relational databases (W3C Technical Report). http://www.w3.org/2004/04/30-RDF-RDB-access/, April 2004.

[20] E. Prud'hommeaux and A. Seaborne. SPARQL Query Language for RDF. W3C Recommendation. http://www.w3.org/TR/2005/WD-rdf-sparql-query-20050217/, February 2005.

[21] V. Raatikka and E. Hyvonen. Ontology-based semantic metadata validation. In *XML Finland 2002 Conference*, Helsinky, Finland, 2002. HIIT Publications.

[22] A. Seaborne and C. Bizer. D2rq – treating non-rdf databases as virtual rdf graphs. In *Proceedings of the 3rd International Semantic Web Conference (ISWC2004)*, 2004.

[23] M. K. Smith, C. Welty, and D. L. McGuinness. OWL Web Ontology Language Guide. http://www.w3.org/TR/2004/REC-owl-guide-20040210/, February 2004.

[24] D. Steer. SquirrelRDF. http://jena.sourceforge.net/SquirrelRDF/.

[25] N. Stojanovic, L. Stojanovic, and R. Volz. A reverse engineering approach for migrating data-intensive web sites to the semantic web. In *IIP 2002*, Montreal, Canada, 2002.

[26] R. Volz, S. Handschuh, S. Staab, L. Stojanovic, and N. Stojanovic. Unveiling the hidden bride: Deep annotation for mapping and migrating legacy data to the semantic web. *Journal of Web Semantics*, 2004.

Publications of the author

Reviewed conference papers

[A.1] M. Švihla [75%], I. Jelínek. Benchmarking RDF Production Tools. Accepted for *18th International Conference on Database and Expert Systems Applications - DEXA '07*, Regensburg, 2007. Lecture Notes in Computer Science by Springer Verlag. To appear.

[A.2] M. Švihla [75%], I. Jelínek. Leveraging Semantic Web Technologies to Integrate Information Related to Academic Papers. Proceedings of eChallenges e-2006, Dublin: IIMC International Information Management Corporation Ltd, 2006, ISBN 1-905824-02-5.

[A.3] M. Švihla [75%], I. Jelínek. The Database to RDF Mapping Model for an Easy Semantic Extending of Dynamic Web Sites. *Proceedings of the IADIS International Conference WWW/Internet 2005 - Volume I*, Lisboa: IADIS Press, 2005, vol. I, pp. 27-35. ISBN 972-8924-02-X.

[A.4] M. Švihla [75%], I. Jelínek. Životný cyklus informácie na sémantickom webe. *DATAKON2005*. Brno: VUT, 2005, pp. 345-355. ISBN 80-210-3813-6.

[A.5] M. Švihla [75%], I. Jelínek. Using Semantic Web Metadata for Advanced Web Information Retrieval. *First International Workshop on Representation and Analysis of Web Space – RAWS-05*, Ostrava - Poruba: VŠB - Technická univerzita Ostrava, 2005, pp. 85-89. ISBN 80-248-0864-1.

[A.6] M. Švihla [75%], I. Jelínek. Improving Web Resources Processing by Distributed Semantic Web Metadata. *E-learning and the Knowledge Society 2005*, Berlin, Germany, 2005, pp. 83-94.

[A.7] M. Švihla [75%], I. Jelínek. Transparent Knowledge Interchange on the Semantic Web. *Proceedings of the International Conference on Computer Systems and Technologies*, Rousse: Bulgarian Chapter of ACM, 2005, pp. 3A 4-1-3A 4-6. ISBN 954-9641-42-2.

Other publications

[A.8] M. Švihla. The Semantic Web. (In Czech) *A chapter in the book New generation of web technologies. Editors M. Bureš, A. Morávek and I. Jelínek.*, WOX, Prague, 2005. 16 pages.

[A.9] M. Švihla [75%], I. Jelínek. Semantic Extension of an University Information Portal. (In Czech) *International Conference on Education, Media, Technology (EMTECH 2005)*, Praha: ČVUT, 2005, ISBN 80-01-03336-8.

[A.10] M. Švihla [75%], I. Jelínek. Metamorphoses - SQL to RDF Mapping for Semantic Web. *18th International Conference on Systems for Automation of Engineering and Research*, Sofia, 2004, ISBN 954-438-428-6.

[A.11] M. Švihla [75%], I. Jelínek. Two Layer Mapping from Database to RDF. *Sixth International Scientific Conference Electronic Computers and Informatics ECI 2004*, Košice, Slovakia, 2004. ISBN 80-8073-150-0.

[A.12] M. Švihla [75%], I. Jelínek. Semantic Web and Automatic Generation of its Content. (In Czech.) *Tvorba softwaru 2004*, Ostrava, 2004. ISBN 80-85988-96-8.

Abstract

The semantic web is the initiative, which strives to solve problems of the current web by adding machine-understandable metadata to web resources. It is a widely-accepted fact that the growth of the semantic web is dependent on the mass creation of metadata that will cover current web resources. In this thesis we aim to address the problem of the semantic web metadata production. Since most web content is backed by relational databases (RDB), this thesis is focused on transformation of relational data into RDF metadata based on mapping between a relational database schema and existing RDFS ontology. This transformation adds explicit semantics to relational data by means of semantic web technologies.

In this thesis we propose a new data transformation model based on two layers. This model is designed with regard to its performance and usability. Our work is built upon the theoretical foundations of semantic web technologies but we also take into account practical issues while developing the formal model. In addition, we implement a data transformation prototype and deploy it in various case studies. The experiments with the implemented system prove that algorithms derived from our data transformation model are more efficient that those published in related works.

The main contributions of the thesis are the following:

1. A formal description of a new model for data transformation based on schema mapping.
2. The design of two declarative XML languages based on the proposed model.
3. The proposal of a high performance data transformation algorithm based on the model.
4. The implementation of the data transformation tool and its deployment in several experimental case studies.
5. A performance analysis of our approach that provides a new perspective on current approaches of RDB to RDF data transformation and RDF storage.

Keywords:

semantic web, data transformation, RDF, relational database, relational model, RDFS, ontology

Anotace

Sémantický web je iniciativa, která se snaží řešit problémy současného webu přidáním strojově srozumitelných metadat ke stávajícím webovým zdrojům. Úspěch této iniciativy závisí mimo jiné na masovém vytváření metadat. Tato práce je zaměřená na problém vytváření metadat sémantického webu. Vzhledem k tomu, že obsah většiny webových zdrojů pochází z relačních databází (RDB), soustřeďujeme se v naší práci na transformaci relačních dat do metadat ve formátu RDF, přičemž je tato transformace založena na mapování mezi relačním databázovým schématem a existující RDFS ontologií. Tato transformace přidává explicitní významy (sémantiku) relačním datům pomocí technologií sémantického webu.

V této dizertační práci představujeme nový model datové transformace. Tento model, skládající se ze dvou vrstev, je navržen s ohledem na výkon a použitelnost. Teoretický model byl implementován ve funkčním prototypu, který jsme nasadili v několika experimentech. Součástí práce jsou taky rozsáhlé výkonnostní testy dokazující, že náš systém je efektivnější než jiné přístupy.

Hlavní přínosy práce:

1. Formální popis nového modelu pro datovou tranformaci, která je založená na mapovaní mezi schématy.

2. Vytvoření dvou XML jazyků pro vyjádření mapování a řízení datové transformace.

3. Návrh rychlého transformačního algoritmu založeného na představeném modelu.

4. Implementace nástrojů pro datovou transformaci a jejich nasazení v několika experimentech.

5. Experimentální ověření výkonu našeho modelu, které přináší nový pohled na současné přístupy k problematice transformace z RDB do RDF a ukládání RDF.

Klíčová slova:

sémantický web, datová transformace, RDF, relační databáze, relační model, RDFS, ontologie