RDB to RDF or Ontology Mapping – Approaches, Tools and Problems[1]

Tatyana I. Ivanova[2]

Abstract: **Web based relational databases are secure, reliable, and widely used, but search engines index only a little part of it content. To facilitate the search and collaborative usage of Web – based scientific data, exporting database schemas as ontologies are needed. This paper explores current approaches and tools for relational databases to ontology translation and mapping to find suitable tool for automatic representation of our scientific data in mashine-processable format.**

Keywords: **database to ontology mapping, relational database to ontology transformation tool, OWL, ontology**

I. INTRODUCTION

Most of data on the Web are stored in relational databases and are accessible for humans through Web browsers. Web applications, as crawling-based general search engines like Google are not fully capable of searching them as their contents, known as deep Web or invisible Web, are hidden behind their Web search interfaces and not effectively crawlable. The form-based interface to relational Web databases, used by humans, is not suitable for intelligent agents. If a software agent was directed to use the e-commerce system for example, it would need to interpret the instructions for filling out the form, enter the appropriate parameters on the form, submit the form, and parse the results. All of the above may be difficult if not impossible for a software agent to do, especially if the requirement is to compare results from multiple e-commerce sites, all with different instructions, forms, and returned data formats. Semantic Web technologies and standards have been developing for making all kind of data, available on a web site or web service, accessible and easy to use from both humans and computers. For that purposes, data semantics have to represent by ontologies and the methods for querying ontologies, return semantic web data formats have to be used. However, relational databases are the best known tools for storing, managing and accessing data, as they are reliable, secure and well-working. Therefore, it is important to establish interoperability between relational databases and ontologies.

Working on our scientific project, we will store valuable testing data in relational database and we wish to make them easily accessible from web for other scientists. For our investigations, comparing our data to another scientific research results is of great importance, and we also would like to simplify finding of related data on the web.

[1] The research, presented in this publication is funded by the Internal Research Project 102ни013-10 Research and Development sector at Technical University of Sofia for 2010

[2]Tatyana I. is from the Technical University of Sofia, Bulgaria , E-mail: tiv72@abv.bg

Effective ways to achieve interoperability between databases are finding mappings between relational database schemas and ontologies-mediators to develop semantic database wrappers, or exporting database schemas as ontologies and dynamically mapping these ontologies. A lot of approaches [3], methods and tools [3] for semantic interoperability of relational databases have been developing during the last years. Because of some significant differences between relational and semantic knowledge models and semantic web technologies immaturity, no one of them can guarantee automatic free of errors disambiguous database to ontology export, or high precision and recall in web searching. We will explore, compare approaches and tools, using described them research papers, documentations, or making our own tests (if systems are available for downloading). Our main aim is to choose the best approaches, methods, find suitable open source and free tools that after some customization we may use to expose our database to semantic web and make possible both it usage from other scientists and finding similar databases in the Web.

II. METHODS FOR EXPOSING RELATIONAL DATABASE DATA TO SEMANTIC WEB

There are grand variety of methods for exposing relational database data to semantic web, differing from each other in used models (annotation or translation), languages and additional database manipulation techniques. Some extract the schema from the database and convert it to semantic web format; others use annotations, or wrappers. Used formats for storing semantic data are usually RDF(S) or OWL. Extracted semantic data can be stored together with database schema in new Semantic Repositories, or remain the relational database, but store Semantic Metadata Extractions in a Separate Repository, or Adding Semantic Markup to the Existing data Repository.

In order to annotate database data, it is necessary to assign the meaningful labels to them. Existing automatic data annotation techniques [10] can be divided into three categories: approaches based on Web interface pages information (for example Arlotta presents an automatic annotating approach which basically annotates data units with the closest labels on result pages), interface schema (DeLa uses some heuristic rules to construct some basic annotators, Yiyao Lu utilizes new features that can be automatically obtained from the result page including the content and data types of the data units to align data), and domain ontology.

The ontology annotation-based approach [2] suppose that database owner produces server side web page markup (usually XML) describing the database's information

structures; The searching client after database location use his own client-side ontology, describing the semantic of his domain and his annotator to produce client-side annotations that conform to his ontology and the server-side markup. Then he can send semantic queries to server-side database, using his ontology and mapping rules through the Web-service API. If there is no server side XML markup, describing the database, client may use deep web crawler for sending random queries to a server to obtain a sample of documents of the underlying collection. The sample represents the entire server content. This representation is called a resource description.

OntoMat-Annotizer is a user-friendly interactive webpage annotation tool (may be used to annotate directly database Web interface forms). It supports the user with the task of creating and maintaining ontology-based OWL-markups i.e. creating of OWL-instances, attributes and relationships. It includes an ontology browser for the exploration of the ontology and instances and a HTML browser that will display the annotated parts of the text. It is Java-based and provides a plug-in interface for extensions. It is freely available and can be downloaded from http://annotation.semanticweb.org/ and come with documentation and tutorial for usage.

Annotation-based approach is easy for database-developers, but he never represents unambiguously the semantic of database content and that sometimes leads to extraction of non-relevant data, or strictly limited reasoning capabilities.

Extraction of the database schema and representing it as ontology ensure semantic access to database data, but there are some problems, related to it. First, there are significant differences in RDB, RDF and OWL models. We will discuss them bellow. Second, manual representation is difficult, time consuming and assume knowledge-management skills, whereas automatic methods are far from his maturity. Third, a lot of web databases are accessible only thought HTML forms, what make additional problems. Extraction of metadata from the database schema is a common method used by OntoKnowledge [5].

Difficulties in database schema extraction methods usage depends from the amount of information about database schema. When we develop our own database, we have full access to it data and schema, and if we plan to make it easily accessible from the web for human and we have to make semantic interface to data by automatic or semiautomatic generation of semantic description as ontology. When we will have an access to not semantically described web database, we can learn about it structure and data only by direct querying through it form-based interface.

There are two main difficulties in translating from RDB to OWL: how to capture and represent all implicitly used in database domain knowledge and how to manage with different logical foundations of RDB and OWL.

Data models, such as database or XML-schemes, typically specify the structure and integrity of data sets, and the semantics of data models often constitute an informal agreement between the developers and the users of the data model and which finds its way only in application programs that use the data model. Ontologies, in contrast, should make explicit all the semantic of data model and make him as much generic and task-independent as possible. There are two main sources for acquisition of all this implied semantic during RDB and OWL mapping process – domain knowledge, represented in machine-processable format as ontologies (automatic approach) and human user or expert (manual approach). Other mismatches between RDB and OWL data models that affect a transformation system are related to inheritance modeling, property characteristics, underlined logical systems and open/closed world assumptions [19].

Deep web crawling and Web information extraction are the two main important areas, related to extracting data and metadata through web interfaces from databases for simplifying the database access. A first prototype deep Web crawler, presented to automatically extract and analyze the interface elements and submit queries through these query interfaces was HiWe. Many independent efforts are proposed for keyword query selection techniques for downloading the textual content from Web repositories.

There has been an active research interest in understanding the semantics of the query interfaces of the structured Web databases [4], [12]. WISE-integrator [12], for example, extracts element labels and default value of the elements to automatically identify matching attributes, [4] uses statistical models to find the hidden domain-specific schema by analyzing co-appearance of attribute names.

Three main models and markups are used for storing extracted knowledge: XML, RDF(S) and OWL. We will discuss them from knowledge representation point of view in details separately lather.

The disadvantage of generating an ontology, based on the database schema approach and converting all the database in a new semantic web format is that any other applications that to interface with the legacy database will need to change. In the tangled network of databases in a corporation or other information organization, this option may be too costly and disruptive to contemplate in the near term. Moreover, response time of the knowledge base strictly depends from his logical model and richness and in some cases checking and querying the base may be very slow (OWL full for example is undesidable, which means that in some cases database couldn't respond within a finite time).

The schema of a database can be extracted and converted into a semantic format such as RDF-S. This semantic version of the schema can be mapped to ontology or published via UDDI or WSDL to make the data available to semantic applications. The semantic metadata and mappings can then be stored in a central repository for the purpose of making queries across multiple data sources.

Semantic markup can be provided at the web page or web service accessing the data or on the repository itself. This approach is used mainly in deep web annotation methods. If a system uses a high level of semantic encoding, there will be greater richness and precision in the semantics available to capture the relationships between concepts that the logical reasoning of agents requires. *Levels of Semantic Encoding* (from lowest level to highest level) are: XML; XML Schema; RDF; RDF-S; DAML + OIL ; OWL Lite ; OWL DL; OWL Full.

There are several tools available for transforming relational databases to ontologies [21], [22], 23]. There are three main approaches, using in transformation: data-mining based, knowledge-based and rule-based. DataGenie [24] is rule-based Protégé's plug-in that is capable of importing data from a relational database and representing it in ontology. This import is simple: each table maps to a class, each column maps to a data type property and each row maps to an instance. The drawback of this simplicity is that DataGenie and similar tools fail to discover inheritance, restrictions, symmetric and transitive properties, object properties and restrictions. It also ignore constraints that capture additional semantics and do not analyze loss of semantics caused by the transformation. The RTAXON learning method combine the most robust rules for exploiting relational schemas with data mining focused on the specific problem of concept hierarchy identification. It is implemented in the RDBToOnto tool, which can be downloading free from [25]. Another similar free java-based tool, RDB2Onto converts selected data from a relational database to a RDF/OWL ontology document based on a defined template. It is intended for ontology population. DB2OWL [14], is another tool for automatic generation of ontologies from database schemas. OntoWrapper [17] exposes external semi-structured data to an ontology repository.

METAmorphoses processor [8] is a tool for the data transformation from a relational database into RDF documents. It is implemented in Java and is based on the two-layer data transformation model: the mapping layer and template layer. In the mapping layer, a given database schema is mapped into a structure of a given ontology. The template layer uses this mapping and produces RDF documents in the way driven by templates.

III. Discovering Mappings Between Relational Database Schemas And Ontologies

The difference between transformation of relational databases to ontologies and database-to-ontology mapping is that the transformation generates ontology, corresponding to database schema, whereas mapping assumes the existence of both a relational database and ontology and produces a set of correspondences between the two. Two main logical models of semantic data representation are used in the Web: RDF – based (including RDF and RDF Schema) and OWL-based (including OWL Lite, OWL DL, OWL Full, and their extensions).

The RDF data model is a directed labeled graph, which consists of nodes and labeled directed arcs linking pairs of nodes. RDF is more expressive than the relational data model and data represented in RDF can be interpreted, processed and reasoned over by software agents. Two main approaches for mapping generation between RDB and RDF are used: Automatic domain-independent Mapping Generation, and Domain Semantics driven Mapping Generation.

Automatic Mapping usually generate mappings between RDB and RDF with RDB table as a RDF class node and the RDB column or relation names as RDF predicates Even though these automatically generated mappings often do not capture complex domain semantics that are required by many applications, these mappings can serve as a useful starting point to create more customized, domain specific mappings, or enable Semantic Web applications to query RDB sources.

The Domain Semantics driven Mapping Generation approach generates mappings from RDB to RDF by incorporating domain semantics that is often implicit or not captured at all in the RDB schema. The explicit modeling of domain semantics, often modeled as domain ontology, enables software applications to take in mind valuable facts or relations, concerning data, that users implicitly assume working with database. There are freely available ontologies in the internet (such as the National Center for Biomedical Ontologies (NCBO) at http://bioportal.bioontology.org/; Gene Ontology GO and so. on.) in almost all domains, that may by used (usually after customization).

The mappings between RDB and RDF may be represented as XPath rules in a XSLT stylesheet, in a XML-based declarative language such as R2O [5], D2RQ [16], D2R MAP[1] or as "quad patterns" defined in Virtuoso's [6] metaschema language. The mappings, especially if they are created by domain experts or reference domain ontology, may have wider applicability.

Mapping of RDB to RDF may be either a static Extract Transform Load (ETL) implementation (called "RDF dump"), and implemented in almost all such tools, or a query-driven dynamic implementation. The dynamic approach, (for example in D2RQ, or Virtuoso systems) implements the mapping dynamically in response to a query.

Tools from the OntoKnowledge project [17] and KAON project [9] can be used for mapping a database schema to an existing ontology or generating an ontology based on the database schema.

Virtuoso RDF View [6] uses the unique identifier of a record (primary key) as the RDF object, the column of a table as RDF predicate and the column value as the RDF subject in the mapping process. Other similar tools are D2RQ [16] and SquirrelRDF [Seaborne et al., 2007]. D2RQ platform is freely available and can be downloading from http://sourceforge.net/projects/d2rq-map/. SquirrelRDF provides access to relational databases, by providing a SPARQL interface to a non-RDF store by extending the basic ARQ - query engine for Jena. This approach ensures a full SPARQL implementation over the foreign data source. SquirrelRDF is freely available and can be downloading from http://sourceforge.net/projects/jena/files/. Triplify [18] is an approach to publish RDF and Linked Data from relational databases. It transforms the resulting relations into RDF statements and publishes the data on the Web in RDF serializations, as Linked Data. Triplify can be easily integrated and deployed with Web applications. It is complemented by a library of configurations for common relational schemata and a REST enabled data source registry.

Creating mappings between database schema and Web ontology is a preconditioning process in the generation of ontological annotations for dynamic Web page contents extracted from the database.

In OWL, a class can be mapped to a relational table. Properties of a class can be mapped to the attributes of a relational table. Inheritance (*subClassOf*) relation between classes can be realized by the foreign key (acting as a primary

key) between relational tables, and foreign key, disjoint with primary key can be mapped to object property. Declarative Languages as D2R MAP may be used to describe mappings between relational database schemata and OWL ontologies, or mappings may be stored as pare of initial ontology [15]. Two main approaches may be used to discover semantic mappings: statistical and knowledge-based.

The mapping process, based on statistical approach [13] starts with a relational schema and an ontology, constructs virtual documents for the entities in the relational schema and the ontology to capture their implicit semantic information, discovers simple mappings between entities by calculating the confidence measures between virtual documents via the TF/IDF model, uses mappings between relations and classes to validate the consistency of mappings between attributes, and properties a set of simple discovered mappings.

Knowledge-based approaches use knowledge sources as WordNet or previously developed domain ontologies for extracting shared concepts between RDB and ontology. These approaches are semiautomatic, or complementary to rule-based and statistical, as the quality of knowledge processing is relatively low.

IV. Discussion And Conclussions

As shown above, we have to expose the schema of our scientific data, using XML – based syntaxes for easy usage from the Web. We may do this, using two different approaches: expose the whole database schema or only propose annotations. Making annotations is easy and they can be easily used from the Web software, but such type of data representation lack of formal semantic and natural language ambiguity can become an obstacle to the proper use of data.

Database schema representation as ontology provides both metadata, related to our data, and formal semantic, and make possible for software agents reasoning about the semantic of the data. Moreover, this representation will be used when we search related to our data in the Web. We will map our ontology concepts to metadata or concepts, representing considered Web databases. Only database schema representation ontology may not be sufficient for performing this mapping process and we will expose additionally our domain ontology (for handling synonymy, one to many domain relationships…) and local context ontology (for explicit representation of all the domain knowledge, implicitly implied, but not explicitly represented in the database), used in the process of automatic building of database ontology. Moreover, queries using semantic web query languages can be imposed to our database through its connection to ontology.

RDBToOnto is a free open source tool for automatically generation of fine-tuned ontologies from relational databases. We plan to use it (may be after some customization) for automatic exporting our database schema to ontology.

Using Protégé, we were able to map ontology instances into relational databases and retrieve results by semantic web query languages. The key idea is that, instead of storing instances along with the ontology terminology, we can keep them stored in a database and maintain a link to the dataset. VisAVis is a open source java-based Protégé plug-in for mapping ontologies to databases, can be download from [15] VisAVis maps the relational database contents to the TBox of the ontology. We plan to use it (may be after some customization) for mapping of external database data to our ontology in the process of searching the related to ours data in the Web.

V. References

[1] C. Bizer, “D2R MAP – A Database to RDF Mapping Language”, In Proceedings 12th International WWW Conference, 2003

[2] S. Handschuh and R. Volz , “Annotation for the Deep Web”, EEE Intelligent Systems archive. Vol. 18 , Issue 5 , 2003)

[3] S.S.Sahoo et al., “A Survey of Current Approaches for Mapping of Relational Databases to RDF”,W3C RDB2RDF Incubator Group, 2009

[4] Z. Zhang, B. He, K. C. Chang, “Understanding Web Query Interfaces: Best-Effort Parsing with Hidden Syntax”,SIGMOD, 2004.

[5] J. Barrasa, A. GómezPérez,, “Upgrading relational legacy data to the semantic web”, In Proc. of 15th international conference on World Wide Web Conference (WWW 2006), pages 1069-1070

[6] C. Blakeley, “RDF Views of SQL Data (Declarative SQL Schema to RDF Mapping)”, OpenLink Software, 2007.

[7] http://www.searchtools.com/info/database-search.html

[8] http://metamorphoses.sourceforge.net/METAmorphoses_processor/

[9] Karlsruhe Ontology Project (KAON). Online. Internet. 2/15/2005. Available at:http://www.KAON.SemanticWeb.org.

[10] C. Xiao-Jun, P. Zhi-Yong, W. Hui, "Multi-source Automatic Annotation for Deep Web," csse, vol. 4, pp.659-662, 2008

[11] S. Handschuh, R. Volz, S.Staab, “Annotation for the Deep Web”, IEEE Intelligent Systems, September/October 2003.

[12] H. He, W. Meng, “WISE-Integrator: A System for Extracting and Integrating Complex Web Search Interfaces of the Deep Web”, VLDB'03, pp.357-368, Berlin, Germany, 2003

[13] W. Hu and Y. Qu, “ Discovering Simple Mappings Between Relational Database Schemas and Ontologies”, 2007, http://dit.unitn.it/~p2p/RelatedWork/Matching/Discovering.pdf

[14] N. Cullot, R. Ghawi, and K.Yétongnon, “DB2OWL: A Tool for Automatic Database-to-Ontology Mapping”, CiteSeerX, 2008

[15] N. Konstantinou, et al. “VisAVis: An Approach to an Intermediate Layerbetween Ontologies and Relational Database Contents http://www.cn.ntua.gr/~nkons/essays_en.html#t

[16] C. Bizer, R.Cyganiak, “D2RQ — Lessons Learned”, W3C Workshop on RDF Access to Relational Databases, 2007.

[17] “The OntoKnowledge Toolset,” Online. Internet., 2004. http://www.ontoknowledge.org/tools/toolrep.shtml.

[18] S. Auer et al., “Triplify Lightweight Linked Data Publication from Relational Database”, WWW 2009, Madrid, Spain

[19] SS Bhowmick, J. Küng, and R. Wagner, “Translating SQL Applications to the Semantic Web” LNCS 5181, pp. 450–464, 2008.

[20]Ontostudio web site http://www.ontoprise.de/en/home/products/ontostudio

[21]. M. Li, X. Du, S.Wang, “Learning Ontology from Relational Database”. ICMLC, Vol. 6, 2005

[22]. G.Shen, et al.,. “Research on the Rules of Mapping from Relational Model to OWL”. Workshop on OWL: Experiences and Directions.Vol. 216 (2006)

[23] I.Astrova, A. Kalja, “Towards the Semantic Web: Extracting OWL Ontologies from SQL Relational Schemata” IADIS International Conference WWW/Internet(2006) 62–66

[24] DataGenie: http://protege.cim3.net/cgi-bin/wiki.pl?DataGenie, 2007

[25] RDBToOnto download page, http://www.tao-project.eu/researchanddevelopment/demosanddownloads/RDBToOnto.