A Survey of Current Approaches for Mapping of Relational Databases to RDF

W3C RDB2RDF Incubator Group January 08 2009

This version:

http://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF_SurveyReport_01082009.pdf

Latest version:

http://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF_SurveyReport.pdf

Authors:

Satya S. Sahoo, Kno.e.sis Center, Wright State University

Wolfgang Halb, Joanneum Research

Sebastian Hellmann, University of Leipzig

Kingsley Idehen, OpenLink Software

Ted Thibodeau Jr, OpenLink Software

Sören Auer, University of Leipzig

Juan Sequeda, University of Texas at Austin

Ahmed Ezzat, Business Intelligence Software Division, HP

Abstract

This document surveys current techniques, tools and applications for mapping between Relational Databases (RDB) and the Resource Description Framework (RDF). Basic knowledge of RDF as well as RDB

concepts and technologies is assumed for readers of this document. The survey is intended to enable the members of the W3C RDB2RDF Incubator Group to:

1. Collate the existing state of the art in mapping approaches between RDB and RDF
2. Use the reference framework defined in this survey to effectively compare the different mapping approaches
3. This survey is a deliverable of the W3C RDB2RDF Incubator Group

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document.

This document is currently a living document and work in progress. Major revisions of this document have to be expected. This document is being developed by the W3C RDB2RDF Incubator Group, part of the W3C Incubator Activity. It represents the consensus view of the group, in particular the authors of this document and those listed in the acknowledgments, on the issues regarding the current state of the art in RDB2RDF mapping approaches.

Publication of this document by W3C as part of the W3C Incubator Activity indicates no endorsement of its content by W3C, nor that W3C has, is, or will be allocating any resources to the issues addressed by it. Participation in Incubator Groups and publication of Incubator Group Reports at the W3C site are benefits of W3C Membership.

Incubator Groups have as a goal to produce work that can be implemented on a Royalty Free basis, as defined in the W3C Patent Policy. Participants in this Incubator Group have made no statements about whether they will offer licenses according to the licensing requirements of the W3C Patent Policy for portions of this Incubator Group Report that are subsequently incorporated in a W3C Recommendation.

Table of Contents

1. Introduction

A critical requirement for the evolution of the current Web of documents into a Web of Data (and ultimately a Semantic Web) is the inclusion of the vast quantities of data stored in Relational Databases (RDB). The mapping of these vast quantities of data from RDB to the Resource Description Framework (RDF) (http://www.w3.org/RDF/) has been the focus of a large body of research work in diverse domains and has led to the implementation of generic mapping tools as well as domain-specific applications. Furthermore, the role of RDF as an integration platform for data from multiple sources, primarily in form of RDB, is one of the main motivations driving research efforts in mapping RDB to RDF. This survey documents current approaches in mapping RDB to RDF and effectively categorizes and compares the different approaches using a well-defined reference framework.

1.1 Mapping between RDB and RDF

The majority of data underpinning the Web and in domains such as life sciences [NCBI resources] and spatial data management [Green et al., 2008] are stored in RDB with their proven track record of scalability, efficient storage, optimized query execution, and reliability. As compared to the relational data model, RDF is more expressive and data represented in RDF can be interpreted, processed and reasoned over by software agents. In "Relational Databases on the Semantic Web" Tim Berners Lee [Berners Lee, 1998] discusses the common and distinct characteristics of RDF and the Entity Relationship (ER) model. The modeling of relationships or predicates in RDF as first class objects is listed as a significant difference between ER and RDF models. The use of Uniform Resource Identifiers (URI) for entities along with the ability to link them together using predicates enables RDF to effectively integrate data from multiple sources.

One of the important objectives of our survey is to analyze the "information gain" of mapping RDB to RDF through explicit modeling of relationships between entities that are either implicit or non-existent in RDB and incorporation of "domain semantics". The incorporation of domain semantics through the use of an application-specific domain ontology in mapping RDB to RDF is an important aspect to fully leverage the expressivity of RDF models. The survey also reviews the use of inference rules over the RDF repository leading to knowledge discovery.

1.2 The RDB2RDF Incubator Group

The W3C launched the RDB2RDF incubator group to explore the issues involved in mapping RDB to RDF in March 2008 with two objectives [W3C XG announcement]:

1. To examine and classify the existing approaches in mapping between RDB and RDF as well as explore the possibility of standardization of the mapping process
2. To examine and classify the approaches in associating OWL classes to SQL queries

This survey was undertaken to partially fulfill the objectives of the incubator group and has been identified as a deliverable along with the final report.

2. Reference Framework for Survey of RDB2RDF Mapping Approaches

A reference framework will enable the effective categorization and comparison of different approaches used in mapping RDB data to RDF. We have defined a reference framework for this survey consisting of six components such as the approach to generate the RDB to RDF mapping, the representation and the achieved integration of data.

2.1 Components of Survey Framework

In this section we describe the individual components of the reference framework in detail:

1. **Creation of Mappings**: We can classify the methods used to generate the mappings between RDB and RDF into two categories:

a) **Automatic Mapping Generation**: [Berners Lee, 1998] discusses a set of mappings between RDB and RDF namely:

i) *A RDB record is a RDF node*

ii) *The column name of a RDB table is a RDF predicate*

iii) *A RDB table cell is a value*

Many systems leverage these mappings to automatically generate mappings between RDB and RDF with the RDB table as a RDF class node and the RDB column names as RDF predicates. An example of this approach is the Virtuoso RDF View [Blakeley, 2007] that 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. Other examples of similar tools are D2RQ [Bizer et al., 2007] (D2RQ also allows users to define customized mappings) and SquirrelRDF [Seaborne et al., 2007].

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. This approach also enables Semantic Web applications to query RDB sources where the application semantics is defined in terms of the RDB schema. This approach is also called "Local ontology mapping".

Reuse of Existing Ontology Schema:

A variation of the automatic generation of mappings between RDB and RDF is the use of an existing ontology to create simple mappings [Hu et al., 2007]. These simple mappings are checked for consistency and subsequently more contextual mappings are constructed. Though this approach automatically generates the RDB to RDF mappings, an existing ontology is used to enhance the quality of the mappings.

b) **Domain Semantics-driven Mapping Generation**: The second 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 a domain ontology, in the RDF repository enables software applications to take advantage of this "information gain" and execute queries that link together entities such as "*gene→expressed_in→brain*" [Sahoo et al., 2008]. Additionally, a mapping generated by using domain semantics also reduces the creation of triples for redundant or irrelevant knowledge. Byrne [Byrne, 2008] discusses the reduction in the size of the RDF dataset by about 2.8 million triples through use of domain semantics-driven mappings from RDB to RDF.

The domain ontology may be pre-existing and sourced from public resources such as the National Center for Biomedical Ontologies (NCBO) at http://bioportal.bioontology.org/ or may be bootstrapped from local ontologies created by automatic mapping tools (as described in previous section (a)). Green et al. [Green et al., 2008] discuss an approach of mapping spatial data to RDF using a hydrology ontology [Hart et al., 2007] as the reference knowledge model and Sahoo et al. discuss an approach to generate mappings using the Entrez Knowledge Model (EKoM) to map gene data to RDF [Sahoo et al., 2008].

This approach is also called "Domain ontology mapping" and the process is the same as an ontology population technique where the transformed data are instances of the concepts defined in the ontology schema. Many mapping tools such as D2RQ [Bizer et al.] allow the users to create customized mapping rules in addition to the automatically generated rules.

2. **Mapping Representation and Accessibility of Mappings**: 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 [Barrasa et al., 2006] or as "quad patterns" defined in Virtuoso's [Blakeley, 2007] meta-schema language. The mappings, especially if they are created by domain experts or reference a domain ontology, may have wider applicability. Hence, to encourage reuse the mappings should be easily accessible by the wider community. Consequently, we review the representation and accessibility of the mappings between RDB and RDF.
3. **Mapping Implementation**: The actual mapping of RDB to RDF may be either a static Extract Transform Load (ETL) implementation or a query-driven dynamic implementation. The ETL implementations such as the application described in Byrne [Byrne, 2008], also called "RDF dump", use a batch process to create the RDF repository from RDB using mapping rules. The dynamic approach, an example application is described in Green et al. [Green et al., 2008], implements the mapping dynamically in response to a query.

   Similar to the data warehouse implementations the ETL approach may not reflect the most current data, but it allows additional processing or analysis over the data including execution of inference rules (well-defined RDF entailments or user-defined rules) without compromising query performance. On the other hand, the dynamic approach has the advantage that the query is executed against the most current version of the data, which assumes significance if the data is frequently updated. However, a dynamic mapping implementation may incur query performance penalties if entailment rules are applied to the RDF repository to infer new information.
4. **Query Implementation**: Queries in systems mapping RDB to RDF may either be in SPARQL, which is executed against the RDF repository, or the SPARQL query may be transformed into one or more SQL queries that are executed against the RDB. Cyganiak [Cyganiak, 2005] has discussed the transformation of SPARQL to relational algebra and further into SQL. This paper describes operators

such as “selection” and “inner join” implemented over RDF and correlates “RDF relational algebra” to SQL.

5. **Application Domain**: As discussed earlier in the “Creation of Mappings” section, an important aspect of mapping RDB to RDF is the incorporation of domain semantics in the resulting RDF. Hence, we explicitly list the application domain of the work reviewed in this survey where possible (mapping tools are not domain-specific).
6. **Data Integration**: The RDF representation model through use of URI and explicitly modeled relationships between entities makes it easier to effectively integrate data from disparate, heterogeneous data sources. It is important to note that RDF does not automatically reconcile the multiple types of heterogeneity, such as structural, syntactic and semantic heterogeneities, that are described in the data/information integration research field. But, the use of domain ontologies along user-defined inference rules for reconciling heterogeneity between multiple RDB sources is an effective integration approach for creating a single or a set of “compatible” RDF. Hence, this metric reviews the different mapping approaches with respect to data integration.

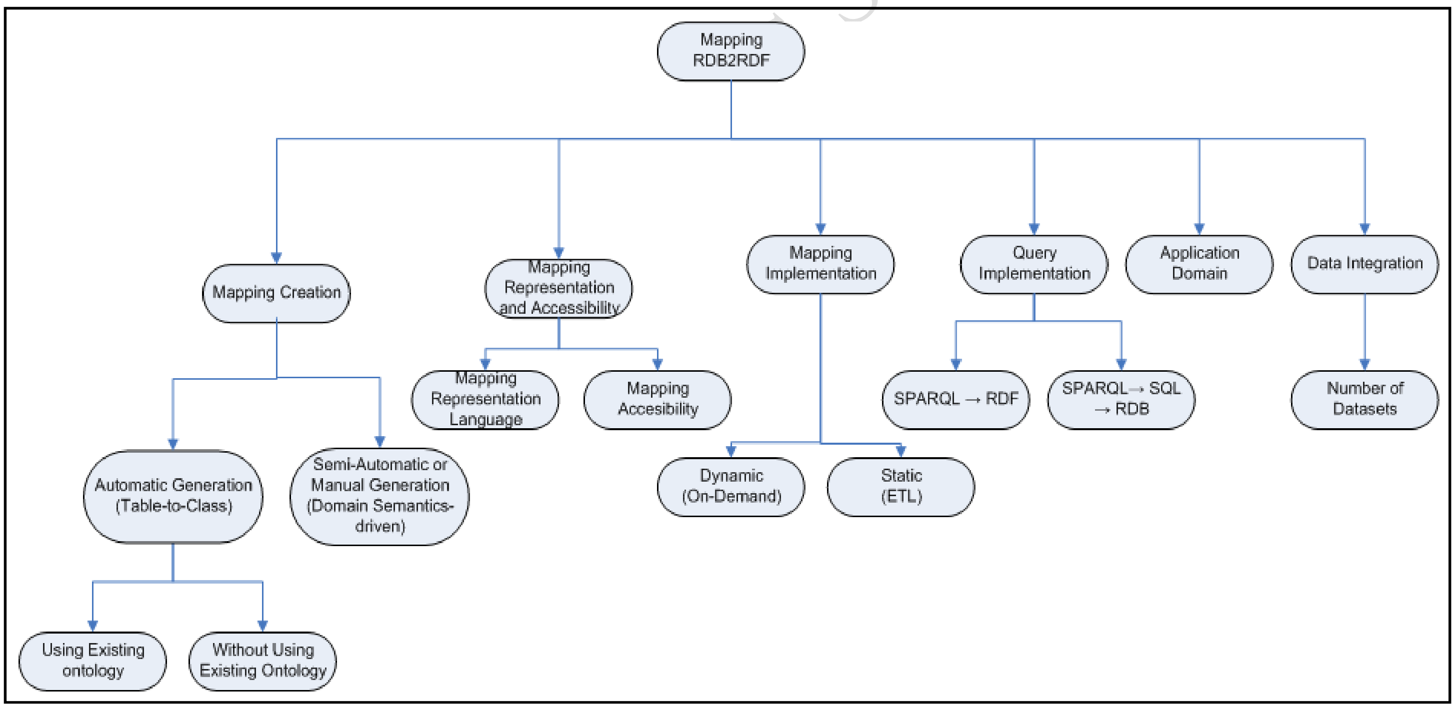

Figure 1: Reference Architecture for RDB2RDF Survey

3. Survey of RDB to RDF Mapping Approaches

In this section, we categorize the surveyed work into three broad classes namely:

1. **Proof of concept projects**: Projects reviewed in this section explore specific approaches to map RDB to RDF with either a prototype or proof of concept implementation. The work may or may not have lead to the release of a tool/application to the community
2. **Domain-specific projects**: Many projects surveyed in this paper were driven by real-world application requirements and have used domain-semantics based customized mappings, generic mapping tools or a combination of both
3. **Tools/Applications**: The projects surveyed in this section include D2RQ, R2O, Virtuoso, Triplify and Dartgrid tools that have been released to the community for mapping RDB to RDF

All the projects in the three categories are reviewed according to the reference architecture defined in Section 2.

3.1 Proof of Concept Projects

1. Hu et al. [Hu et al., 2007] describe an approach that aims at the automatic creation of simple mappings between relational database schemas and ontologies. Based on the relational schema and an ontology, initial simple mappings are derived and then checked for consistency. Based on sample input (mappings and instances for both the relational schema and the ontology) more contextual mappings are constructed. Experimental results in a limited domain showed the feasibility of the approach.
2. Kashyap et al. [Kashyap et al., 2007] describe their work involving a mediator based approach to represent mappings from ontological concepts to disparate data sources as part of a general framework for RDF based access to heterogeneous data sources. The heterogeneous data sources, illustrated using a life sciences domain scenario, include RDB, Web services and Excel sheets (using the MS Office API). The SPARQL queries are automatically translated to the appropriate query language using the mappings represented by the mediator classes.
3. Cullot et al. [Cullot et al., 2007] describe DB2OWL using the table to class and column to predicate approach but use specific relational database schema characteristics, that is, how tables relate to each other, to assert subclass and other object properties. The object properties represent many-to-many relationships and referential integrity. The mappings are stored in a R2O document.
4. Tirmizi et al. [Tirmizi et al., 2008] follow Li et al. [Li et al., 2005] and similar work. It is the first work to present formal rules in First Order Logic to translate table to class and column to predicate. A notion of completeness for a transformation system is also presented based on all the possible foreign key and primary key combinations.

5. The Semi-automatic Ontology Acquisition Method (SOAM) work by Li et al. [Li et al., 2005] uses the table to class and column to predicate approach to create an initial ontology schema which is then refined by referring to a dictionary or thesauri (for example, WordNet). Constraints in the relational model are mapped to constraints in the ontology schema. For example, "NOT NULL" and "UNIQUE" are mapped to cardinality constraints on the respective properties. If a given set of relations is mapped to an ontology concept, the corresponding tuples of the relations are transformed to instances of the ontology concept.

3.2 Domain-specific Projects

1. Sahoo et al. [Sahoo et al., 2008] describe work in the life sciences domain that incorporates domain semantics (from multiple, integrated ontologies) to create the mappings (represented as XPath rules in XSLT stylesheet) from RDB to RDF. A RDF dump is created using a batch approach and stored in Oracle 11g. SPARQL is used to query the RDF repository.
2. Byrne [Byrne, 2008] describes an application in the cultural heritage domain to convert the National Monument Record of Scotland data stored in a relational database to RDF. The Simple Knowledge Organization System (SKOS) framework is used to transform the Royal Commission on the Ancient and Historical Monuments of Scotland (RCAHMS) thesauri to the Semantic Web. The entire RCAHMS database with 1.5 million entities is converted in to 21 million RDF triples and implemented on both Jena and AllegroGraph systems.
3. Green et al. [Green et al., 2008] describe the integration of spatial data in RDB for predictive modeling of diffuse water pollution using OWL-DL ontologies at multiple levels. The ontologies at the first level (called *data ontologies*) are used to map each data source to concepts in the ontologies at the next level (called *domain ontologies*). The "data ontologies" are represented in the D2RQ mapping language. The "application ontology", at the third level, links the "domain ontologies" and also adds application-specific information. The D2RQ engine is modified to include spatial operators and is used to interface between data sources and data ontologies. The SPARQL query language is used to query the virtual RDF graphs generated from the data sources.

3.3 Tools/Applications

1. The Virtuoso RDF View [Blakeley, 2007] uses the table to class (RDFS class) and column as predicate approach and takes into consideration special cases such as whether a column is part of the primary

key or foreign key. The foreign key relationship between tables is made explicit between the relevant classes representing the tables.

The RDB data are represented as virtual RDF graphs without physical creation of RDF datasets. Virtuoso RDF views are composed of "quad map patterns" that defines the mapping from a set of RDB columns to triples. The quad map pattern is represented in the Virtuoso meta-schema language, which also supports SPARQL-style notations.

2. D2RQ [Bizer et al., 2007] provides an integrated environment with multiple options to access relational data including "RDF dumps", Jena and Sesame API based access (API calls are rewritten to SQL), and SPARQL endpoints on D2RQ Server.

   The mappings may be defined by the user thereby allowing the incorporation of domain semantics in the mapping process, although there are some limitations to this as described in the Ordnance Survey presentation [Greene et al., 2008]. The mappings are expressed in a "declarative mapping language". The performance varies depending on the access method and is reported to perform reasonably well for basic triple patterns but suffers when SPARQL features such as FILTER, LIMIT are used.

3. Triplify [Auer, 2005] is a simplistic approach to publish RDF and Linked Data from relational databases. Triplify is based on mapping HTTP-URI requests onto relational database queries expressed in SQL with some some additions. Triplify transforms the resulting relations into RDF statements and publishes the data on the Web in various RDF serializations, in particular as Linked Data. Triplify as a light-weight software component, which can be easily integrated and deployed with the numerous widely installed Web applications. The approach does not support SPARQL, includes a method for publishing update logs to enable incremental crawling of linked data sources. Triplify is complemented by a library of configurations for common relational schemata and a REST-enabled datasource registry. Despite its lightweight architecture Triplify is usable to publish very large datasets, such as 160 GB of geo data from the OpenStreetMap project.

4. R2O [Barrasa et al., 2006] is a XML based declarative language to express mappings between RDB elements and an ontology. R2O mappings can be used to "detect inconsistencies and ambiguities" in mapping definitions. The ODEMapster engine uses a R2O document to either execute the transformation in response to a query or in a batch mode to create a RDF dump.

5. Wu et al. [Wu et al., 2006] and Chen et al. [Chen et al., 2006] describe the Dartgrid Semantic Web toolkit that offers tools for the mapping und querying of RDF generated from RDB. The mapping is basically a manual table to class mapping where the user is provided with a visual tool to define the

mappings. The mappings are then stored and used for the conversion. The construction of SPARQL queries is assisted by the visual tool and the queries are translated to SQL queries based on the previously defined mappings. A full-text search is also provided.

The tool is available at: http://ccnt.zju.edu.cn/projects/dartgrid/

6. The RDBToOnto work by Cerbah et al. [Cerbah, 2008] is a highly configurable tool that eases the design and implementation of methods for ontology acquisition from relational databases. It is also a user-oriented tool that supports the complete transitioning process from access to the input databases to generation of populated ontologies. The settings of the learning parameters and control of the process are performed through a full-fledged dedicated interface.

   The tool is available at: http://www.tao-project.eu/researchanddevelopment/demosanddownloads/RDBToOnto.html
7. Asio Semantic Bridge for Relational Databases (SBDR) and Automapper: use the table to class approach. Automapper generates an OWL Full ontology from a relational database. In the generated ontology, each class corresponds to a table in the relational database and columns are represented as properties of the relevant class. A primary key column has cardinality set to 1. A nullable column has max cardinality set to 1. For a foreign key, an object property is created and its range is set to the corresponding class. The generated ontology includes SWRL rules to equate individuals based on multiple primary key columns. SBDR provides an RDF view of data in the relational database. SPARQL queries can be written in terms of the Automapper generated data source ontology and relational data is returned as RDF. SBRD rewrites the SPARQL query to SQL, executes the SQL and converts SQL rows to RDF conforming to the data source ontology.

PROJECTS	MAPPING CREATION	MAPPING REPRESENTATION AND ACCESSIBILITY		MAPPING IMPLEMENTATION	QUERY IMPLEMENTATION	APPLICATION DOMAIN	DATA INTEGRATION	
	Automatic (Table-to-Class) **or** **Manual/Semi-Automatic** (Domain Semantics-driven)	**Representation Language**	**Mapping Access**	**Static (ETL) or Dynamic**	**SPARQL → RDF or SPARQL→SQL→ RDB**		**Yes/No**	**Number of Datasets**
1. Hu et. al, 2007	Automatic (with use of existing ontology)	First Order Logic formulae or Horn Clauses	Files	None Specified	None Specified	Generic	Enables (through contextual mappings)	Potentially Multiple
2. Kashyap et al, 2007	Manual/Semi-Automatic (Domain Semantics-driven)	Mediator Framework Classes	Mapping mediator	Dynamic	SPARQL→SQL→ RDB	Life Sciences	Enables	Potentially Multiple
3. DB2OWL (Cullot et al., 2007)	Automatic (Table to Class)	R2O language	R2O mapping document		SPARQL→SQL→ RDB	Generic	Enables	Potentially Multiple
4. Tirmizi et. al 2008	Automatic (Table to Class, SQL-DDL to RDF)	First Order Logic	None specified	Static	None Specified	Generic	No	None
5. SOAM (Li et al, 2005)	Automatic (Table to Class) with user input	Logic Rules	Implemented as part of system	Static	Potentially SPARQL (on generated populated ontology)	Generic (Case Study: Economics)	No	None
6. Sahoo et al., 2008	Manual/Semi-Automatic (Domain Semantics-driven)	XPath expressions	XSLT document	Static	SPARQL	Life Sciences	Yes	Test included five (Gene, Biological Pathway)
7. Byrne, 2008	Manual/Semi-Automatic (Domain Semantics-driven)	SKOS vocabulary	RDF document	Static	SPARQL	Cultural Heritage	No	None
8. Green et. al, 2008	Manual/Semi-Automatic (Domain Semantics-driven)	D2RQ language	D2RQ mapping file	Dynamic	SPARQL→SQL→ RDB	Ordnance Survey	Yes	Mutltiple
9. Virtuoso RDF View (Blakeley, 2007)	Both (user-specified)	SPASQL-based Meta Schema Language	Quad Storage	Both	Both	Generic	Enables	Potentially Multiple
10. D2RQ (Bizer et al., 2007)	Both (user-specified)	D2RQ language	D2RQ mapping file	Both	Both	Generic	Enables	Potentially Multiple
11. Triplify [Auer, 2009]	Manual	SQL	Config File	Both (Linked Data)	Linked Data	Generic	Yes	Multiple
12. R2O (Barrasa et al., 2006)	Both (user-specified)	R2O language	R2O mapping document	Both	Both	Generic	Enables	Potentially Multiple
13. Dartgrid (Wu et al., 2006)	Automatic (Table to Class)	XML File	Visualized Mapping tool	Dynamic	SPARQL→SQL→ RDB (Provide search and query interface)	Life Science (Traditional Chinese Medicine, TCM)	Yes	Test included databases for herb, compound formulas, disease, drug, TCM treatment.
14. RDBtoOnto (Cerbah, 2008)	Automatic (Table to Class, allows user intervention)	Constraint rules	Not explicitly stored	Static	Potentially SPARQL (on generated populated ontology)	Generic	No	None
15. Asio Tools	Automatic (Table to Class)	OWL Full based language	File based	Both	SPARQL→SQL→ RDB	Generic	Enables	Potentially Multiple

Table 1: A Comparative View of Implementation using Survey Reference Framework

4. Discussion

The survey highlights different aspects of the RDB to RDF mapping process; some of these aspects need to be addressed to enable the move towards standardization of the mapping process.

Mapping Representation and Accessibility - As presented in the survey, currently there is no standard method for representation of mappings between RDB and RDF. Projects reviewed in the survey used a variety of representation formats such as FOL, XPath expressions, and tool-specific languages (for example, D2RQ or R2O mapping language). Mappings, especially those that are created by domain experts or use domain ontologies, are important artifacts that should be available for re-use.

Mapping Implementation – The projects reviewed in this survey either implemented a dynamic on-demand or a static mapping from RDB to RDF. Although the advantages and disadvantages of each approach such as accessibility to latest version of data, update patterns of the data sources are well-documented in the data warehouse community, the projects reviewed in this survey did not contain an explicit comparison between the two approaches. The RDB2RDF incubator group has discussed this issue in "Requirements for Relational to RDF Mapping" available at: http://esw.w3.org/topic/Rdb2RdfXG/ReqForMappingByOErling.

5. Conclusion

This survey to document the current state of the art in mapping approaches between RDB and RDF was conducted to partially fulfill the objectives of the W3C RDB2RDF incubator group. To enable a coherent and effective comparison of the different RDB2RDF mapping approaches we defined a reference architecture for the survey consisting of six components such as mapping generation, query execution and data integration achieved by mapping RDB to RDF.

One of the important aspects of mapping RDB to RDF is the potential to explicitly model information that was either implicitly modeled or not represented at all in RDB, such as domain semantics. Hence, many of the tools and generic application in the survey have noted the importance of using a domain ontology, in addition to information from the RDB schema, in generating RDF. Another important aspect of mapping RDB to RDF is the potential for data integration by representing data from multiple RDB sources as a single RDF graph. The representation of mappings between RDB and RDF in a standardized form is necessary to enable their reuse and the RDB2RDF incubator group, in its final report [RDB2RDF XG Final Report] has proposed the use of the W3C Working Group Rule Interchange Format (RIF) (http://www.w3.org/2005/rules/wiki/RIF_Working_Group) to represent mappings. None of the projects reviewed in the survey use RIF to represent mappings between RDB and RDF.

Finally, this survey is expected to not only serve as a resource to the RDB2RDF incubator group but also for the community of researchers involved in mapping RDB to RDF in order to support the evolution of the Web of documents into a "Web of Data".

6. Acknowledgement

We would like to thank other member of the W3C RDB2RDF Incubator Group for the valuable suggestions and comments.

7. References

1. [Berners Lee, 1998], "Relational Databases on the Semantic Web", Berners-Lee, T., 1998, http://www.w3.org/DesignIssues/RDB-RDF.html (retrieved Jan 12, 2009)
2. [Barrasa et al., 2006], "Upgrading relational legacy data to the semantic web", Barrasa,J., Gómez-Pérez, A.,. In Proc. of 15th international conference on World Wide Web Conference (WWW 2006), pages 1069-1070, Edinburgh, United Kingdom, 23-26 May 2006.
3. [Bizer et al., 2007], "D2RQ — Lessons Learned", Bizer C., Cyganiak, R., Position paper for the W3C Workshop on RDF Access to Relational Databases, Cambridge, USA, 25-26 October 2007.
4. [Blakeley, 2007], "RDF Views of SQL Data (Declarative SQL Schema to RDF Mapping)", Blakeley, C., OpenLink Software, 2007.
5. [Chen et al., 2006], "Towards a Semantic Web of Relational Databases: A Practical Semantic Toolkit and an In-Use Case from Traditional Chinese Medicine", Chen, H., Wang, Y., In Proc. of 5th International Semantic Web Conference (ISWC 2006), pages 750-763, Athens, USA, 5-9 November 2006.
6. [Cerbah, 2008], "Learning Highly Structured Semantic Repositories from Relational Databases - The RDBToOnto Tool.", Proceedings of the 5th European Semantic Web Conference (ESWC 2008), Tenerife, Spain, June, 2008.
7. [Cullot et al., 2007], "DB2OWL: A Tool for Automatic Database-to-Ontology Mapping", Cullot, N., Ghawi, R., Yetongnon, K.,. In Proc. of 15th Italian Symposium on Advanced Database Systems (SEBD 2007), pages 491-494, Torre Canne, Italy, 17-20 June 2007.
8. [Green et al., 2008], "Creating a semantic integration system using spatial data", Green, J., Dolbear, C., Hart, G., Engelbrecht, P., Goodwin, J., in *International Semantic Web Conference 2008* Karlsruhe, Germany.

9. [Hu et al., 2007], "Discovering Simple Mappings Between Relational Database Schemas and Ontologies", Hu, W., Qu, Y., In Proc. of 6th International Semantic Web Conference (ISWC 2007), 2nd Asian Semantic Web Conference (ASWC 2007), LNCS 4825, pages 225-238, Busan, Korea, 11-15 November 2007.
10. [Kashyap et al., 2007], "From Web 1.0 -> 3.0: Is RDF access to RDB enough?", Kashyap, V., Flanagan, M., Position paper for the W3C Workshop on RDF Access to Relational Databases, Cambridge, USA, 25-26 October 2007.
11. [Sahoo et al., 2008], "An ontology-driven semantic mash-up of gene and biological pathway information: Application to the domain of nicotine dependence", Sahoo, S.S., Bodenreider, O., Rutter, J., Skinner, K., and A. Sheth. Journal of Biomedical Informatics (Special Issue: Semantic Biomedical Mashups), (in press), 2008.
12. [Wu et al., 2006], "Dartgrid: a Semantic Web Toolkit for Integrating Heterogeneous Relational Databases", Wu, Z., Chen, H., Wang, H., Wang, Y., Mao, Y., Tang, J., Zhou, C., Semantic Web Challenge at 4th International Semantic Web Conference (ISWC 2006), Athens, USA, 5-9 November 2006.
13. [Li et al., 2005], "A Semi-automatic Ontology Acquisition Method for the Semantic Web dblp", Man Li, Xiaoyong Du,Shan Wang,
14. [Tirmizi et al., 2008], "Translating SQL Applications to the Semantic Web", Tirmizi, S., Sequeda, J., Miranker, D., Lecture Notes in Computer Science, Volume 5181/2008 (Database and Expert Systems Applications)
15. [Byrne, 2008], "Having Triplets – Holding Cultural Data as RDF", Byrne, K, Proceedings of the ECDL 2008 Workshop on Information Access to Cultural Heritage, Aarhus, Denmark, September 18, 2008, M. Larson, K. Fernie, J. Oomen, and J. Cigarran, Eds., September 2008.
16. [Hart et al., 2007], Hart, G, Dolbear C and Goodwin, J. (2007) 'Lege Feliciter: Using Structured English to represent a Topographic Hydrology Ontology.' In Proceedings of the OWL Experiences and Directions Workshop, 2007
17. [RDB2RDF XG Final Report] retrieved Jan12,2009, http://www.w3.org/XG_Report/RDB2RDF_XG
18. [NCBI Resources] retrieved Jan12,2009, http://www.ncbi.nlm.nih.gov/
19. [W3C XG announcement], retrieved Jan12,2009, http://www.w3.org/News/2008#item34
20. [Seaborne et al., 2007], "SQL-RDF", Seaborne, A., Steer, D., Williams, S., http://www.w3.org/2007/03/RdfRDB/papers/seaborne.html (retrieved on Jan 12, 2009)
21. [Cyganiak, 2005], "A relational algebra for SPARQL", Cyganiak, R, 2005, HP Technical Report, HPL-2005-170
22. [Auer, 2005] "Triplify - Lightweight Linked Data Publication from Relational Database", Auer, S., Dietzold, S., Lehmann, J., Hellmann, S., Aumueller,D., To appear in proceedings of WWW 2009, Madrid, Spain