I.J. Information Technology and Computer Science, **2015, 01, 48-53**
Published Online December 2014 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2015.01.06

A New OWL2 Based Approach for Relational Database Description

Naïma S. Ougouti
University of Sciences and Technology of Oran-Mohamed Boudiaf (USTO-MB)
Faculty of Mathematics and Computer Science, LSSD Laboratory, Oran, 31000, Algeria
E-mail: Souad.ougouti@univ-usto.dz

Hafida Belbachir
University of Sciences and Technology of Oran-Mohamed Boudiaf (USTO-MB)
Faculty of Mathematics and Computer Science, LSSD Laboratory, Oran, 31000, Algeria
E-mail: h_belbach@yahoo.fr

Youssef Amghar
LIRIS UMR 5205, Insa of Lyon, 69000, France
E-mail: Youssef.amghar@insa-lyon.fr

Abstract—Nowadays, the scientific community is more and more interested by the mediation problem within Peer-to-Peer (P2P) systems and by data sources migration within the semantic web. Data integration and interoperability become a necessity to meet the need for information exchange between heterogeneous information systems. They reflects the ability of an information system to collaborate with other systems sometimes of a very different nature and aims at developing architectures and tools for sharing, exchanging and controlling data. In this context we have proposed a new heterogeneous and distributed data management system in a P2P environment called MedPeer. Among this system functions, we have focused in this article on relational databases description through the use of ontologies. We thus propose Relational.OWL2E, a new approach that, starting from the relational schema, generates an ontology based on the OWL2 language. Our main contribution lies in the semantics we have added to relational databases concepts in representing attributes by rich XML schema datatypes, primary keys, unique keys, foreign keys and by associating to each class a set of synonyms in order to guide the process of discovering semantic correspondences.

Index Terms— Semantic Web, Ontologies, Relational Databases, Web Ontology Language (OWL2), Schema Representation

I. Introduction

Since those past few years we have witnessed the emergence of new applications that need to share information between different systems. This is the case of e-government, e-learning, e-commerce, bioinformatics and electronic libraries. However, in this context, information systems, designed and developed by different organizations, generally constitute heterogeneous and autonomous data sources.

As commerce and computer science are developing rapidly, databases become more widely used and translating data between multiple distributed databases becomes a growing need, so database integration is long standing open problem with extensive research literature [1].

Thus, interoperability has become a necessity to meet the need for information exchange between heterogeneous information systems. It reflects the ability of an information system to collaborate with other systems sometimes of a very different nature and aims at developing architectures and tools for sharing, exchanging and controlling data.

Semantic web and ontologies give solutions for interoperability. The goal of Semantic Web is to add semantics to the existing data on the web and thus create an integrated web of data [2]. Ontologies are very useful in increasing Information Retrieval performance. they deals with occurrence of events, their instances and user defined relations between concepts. This represents background knowledge on Semantic level where Semantic level is defined as set of semantic entities including their concepts and relations instead of simple words which are used in thesaurus [3].

In this context, we have introduced a new data integration system in a P2P environment named MedPeer [4]. It has a Super-peer architecture based on peers regrouping according to media type (Texts, Images, Relational databases, semi-structured...). Super-peers form between them a pure P2P network. This architecture combines a centralized approach with a non structured one thus providing the advantages of centralized research such as autonomy, tasks distribution and robustness for a distributed research. Each super-peer manages the peers containing the same type of media it represents; it is selected according to its calculation capacities and band-width. In addition, it must have all necessary information to be able to direct requests arriving to it towards relevant peers. Semantic mediation is essential because schema sources are different. This function is achieved by a source description module that has for principal goal to regulate peers syntactic and semantic heterogeneity problem in a community. Each peer data source will be described by an ontology using our new approach.

These ontologies will be regularly sent to the super-peer community, to enable it to generate semantic correspondences with domain ontology. All this permits to deal with possible data sources modifications and with system dynamicity.

In this article, we will focus on this latter problem by presenting a new relational schema representation format based on the OWL Web Ontology Language in its second version named Relationnel.OWL2E. By exploiting the different opportunities provided by OWL2 [5]. and our ontology, we are now able to describe and share any relational database schema.

This paper is organized as follows:

In Section 2, we will present a state of the art of the main approaches that describe relational databases with ontologies. In section 3, we will introduce Relationnal.OWL2E our new OWL2 based approach for relational database description. In Section 4, we will illustrate our approach with an example before our conclusion.

II. State of the Art

Wanting to take advantage from the benefits brought by the Semantic Web, several works the goal of which is the passage from a relational database to a newer format (XML / RDF / OWL) have emerged. We have chosen to present six approaches [6][7][8][9][10][11]., more recent methods are in [12][13][14][15].

Relational.OWL [6]. translates the majority of relational model concepts into OWL, from relational schema to data including integrity constraints. This system defines four classes and a set of properties allowing to link them together.

In Table 1 are listed the predefined classes and Table 2 contains the different properties.

The prefixes rdf, rdfs, dbs, xsd and owl represent namespaces used in the Relational.OWL ontology.

Table 1. Classes of Relationnal.OWL

rdf:ID	rdfs:subClassOf	rdfs:comment
dbs: Database	rdf:Bag	The Class of Databases
dbs:Table	rdf:Seq	The Class of Tables
dbs:Column	rdf:Ressource	The class of Databases columns
dbs:PrimaryKey	rdf:Bag	The Primary key of a table

Table 2. Properties of Relational.OWL

rdf:ID	rdfs:domain	rdfs:range	rdfs: comment
dbs:has Table	dbs:Database	dbs:Table	A Database has a set of Tables.
dbs:has Column	dbs:Table	dbs:Column	A Table has a set of Column
dbs:isIdentified By	dbs:Table	dbs:Primary Key	A Table is identified by a Primary Key.
dbs:references	dbs:Column	dbs:Column	Foreign Key rel.ship between Columns
dbs:length	dbs:Column	xsd:nonNegati-ve Integer	Maximal length of an entry in that Column
dbs:scale	dbs:Column	xsd:nonNegati-ve Integer	The scale an entry of the Column may have.

OntoGrate [7]. is a relational database integration system in a P2P (Peer- to-Peer) environment. To represent relational schemas in OWL, the authors have extended the expressiveness of the web ontology language. They thus have introduced a new language, Web-PDDL, an extension of PDDL (Planning Domain Definition Language) based on the logic applied to first order predicates. At first, the database concepts are translated through the use of the Web-PDDL language. Once the ontology generated, the system has a syntax adapter named PDDOWL, which translates the first Web-PDDL ontology into OWL ontology. In the final Generated ontology, a table is transformed into a class, subclass of the class sql: relationship (Defined in OntoGrate system as the class representing tables), an attribute is transformed into an OWL property, a constraint is seen as an axiom (rule) and a primary key constraint as a functional OWL constraint (owl: FunctionalProperty).

RDF Gateway [8]. is a system that translates a relational database schema into RDFS or OWL ontology via the *schema_type* parameter, which specifies the ontology default output.

The SQL Data service is a module of RDF Gateway system that queries the database and extracts the relational schema then transforms it into RDFS or OWL ontologies. In this system a table is translated into a class, an attribute into a property *rdfs:property* for an RDFS output or *owl:DatatypeProperty* for OWL output, a foreign key into a property *rdfs:property or owl:ObjectProperty* and finally the datatype of attributes are translated into XML Schema datatypes.

OWL_K (K for *Key*) [9]. is an extension of OWL to manage identification constraints which are equivalent to primary keys of the relational model. This work was motivated by the difficulties of the OWL DL dialect to capture their semantics. The default vocabulary of OWL was extended to take into account these constraints.

The system proposes:

- *The ICAssertion class* which represents the identification constraint.
- The property *onClass* which is the class (table) on which falls the identification constraint.
- The property byProperty *which* represents a property (attribute) participating to the identification constraint.

The default OWL description logic language has also been extended to take into account the new concepts semantic.

In this system, datatypes are translated by using XML Schema and foreign keys are translated by using cardinality constraints (*owl: minCardinality*, *owl: cardinality*, *owl: maxCardinality*).

"Reference [10]" developed a tool called DB2OWL to create ontology from a relational database. It looks for some particular cases of database tables to determine which ontology component has to be created from which database component. The created ontology is expressed in OWL-DL language which is based on Description Logics. The mapping process starts by detecting some particular cases for tables in the database schema. According to these cases, each database component (table, column, constraint) is then converted to a corresponding ontology component (class, property, relation). The set of correspondences between database components and ontology components is conserved as the mapping result to be used later.

R2O [11]. is an extensible, fully declarative language to describe mappings between relational DB schemas and ontologies. It is intended to be expressive enough to describe the semantics of these mappings. R2O is a RDBMS independent high level language that works with any DB implementing the SQL standard. Its main features are:

1) Its mapping defines how to create instances in the ontology in terms of the data stored in the DB.
2) Its mapping definition can be used to automatically populate an ontology with instances extracted from the DB content and can also be used to automatically characterize data sources to allow dynamic query distribution in intelligent information integration approaches.

III. Relational.OWL2E

The solution we are proposing here is an extension of the Relational.OWL proposed system [6]. We chose this approach because of its specificity to translate almost all the concepts of the relational model in OWL ontologies.

Our main contribution lies in the semantics we have added to relational databases concepts in representing attributes by rich XML schema datatypes, primary keys, unique keys, foreign keys and taking into account the NULL and NOT NULL constraints of the relational model. We have also associate to each class a set of keywords (synonyms) in order to capture the semantics of the terms used to guide the process of discovering semantic correspondences.

We called this ontology Relational.OWL2E because it is based on OWL2 and Relational.OWL that we have extended (E).

We obtain information on the database content from its data dictionary (catalog), and then we generate the corresponding ontology by translating tables, attributes (columns), datatypes (possibly with length restrictions), primary keys, unique keys and foreign keys into ontology concepts.

We thus defined 5 classes and 9 properties between them; they are summarized in the two following tables:

Table 3. Classes in Relational.OWL2E

Classes	Comments
Database	The class of databases
Table	The class of tables
Column	The Class of columns
PrimaryKey	The Class of primary keys
UniqueKey	The Class of Unique keys

Table 4. Properties in Relational.OWL2E

Properties	rdfs:domain	rdfs:range	Comments
Has	owl:Thing	owl:Thing	A thing has another thing.
hasTable	Database	Table	A Database belongs to a set of Tables.
hasColumn	Table PrimaryKey UniqueKey	Column	A Table belongs to a set of Columns.
hasPrimaryKey	Table	PrimaryKey	A Table is identified by a Primary Key
hasUniqueKey	Table	UniqueKey	A table may have unicity constraints on certain attributes
hasForeignKey	Table	Table	A table references another table in a foreign key relation.
References	Column	Column	A column references another column in a foreign key relation.
hassynonym	Database Table Column	Rdfs :litteral	The name of a database, a table or a column may have synonyms
Isa	Table	Table	hierarchical relationship between two tables

A.Relational.OWL2E Ontology Serialization

In what follows we will give a few Relational.OWL2E ontology extracts, in RDF/XML syntax.

Class Definition

<owl:Class rdf:ID="Table" >

<rdfs:subClassOf
rdf:about="http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag"/>
< rdfs:label xml:lang="en">Table</rdfs:label >
<rdfs:comment xml:lang="en">The class of database tables. </rdfs:comment >
</owl:Class >

Property Definition

< owl:ObjectProperty rdf:ID="hasTable" >
< rdfs:subPropertyOf rdf:resource="#has"/>
< rdfs:domain rdf:resource="#Database"/>
< rdfs:range rdf:resource="#Table"/>
< rdfs:label xml:lang="en">hasTable</rdfs:label >
<rdfs:comment xml:lang="en">A Database has a set of tables </rdfs:comment >
</owl:ObjectProperty >

B. Translating algorithm

First part

- Firstly, all tables are extracted from a database.
- The Database name represents a Database class in Relational.OWL2E
- Each table name will be expressed as a table class then as hasTable property value.
- From each table are extracted attributes, primary, unique and foreign keys.
- Each attribute name will be expressed as a *hasColumn* property value.
- The Primary key will be expressed by the *hasPrimaryKey* property on the *PrimaryKey* class containing the list of attributes participating in the key, each attribute being expressed as a *hasColumn* property value.
- The Unique key will be expressed similarly as the primary key, but with the *hasUniqueKey* property on the *UniqueKey* class containing the list of attributes participating in the key, each attribute being expressed as a *hasColumn* property value.
- The Foreign key will be expressed by the *hasForeignKey* property. This property value will be the table referenced by the foreign key. Each foreign key attribute will be expressed as a *Column* class instance and linked to the referenced column by the *references* property having for value the referenced attribute column.

The result of this algorithm first part will be an ontology that describes all the database schema concepts. Attributes datatypes are treated in the algorithm second part.

Second Part

1) Each attribute will be expressed as a *datatype property* , whose domain (*rdfs:domain*) is the name of the class representing the table containing the attribute and its image (*rdfs:range).* Its datatype is expressed with XML Schema datatypes in the following way:
   - Integer is expressed by the XML Schema integer datatype, with possible restrictions on datatype values intervals, thanks to the XML schema facets *maxInclusive*, *maxExcusive*, *minInclusive*, *minExclusive*
   - Decimals are expressed by the decimal XML schema datatype, with possible restrictions, thanks to the *totalDigits* and *fractionDigits* facets.
   - String will be expressed by the string XML schema datatype. We use *minLength* and *maxLength* facets to express the minimum and maximum number of characters allowed. For the *minLength* facet value, if the attribute accepts null values, then minLength will be 0, otherwise 1.
   - The Set datatype, is translated into a string datatype, its *maxLength* facet value will be extracted from the MySQL catalogue.
   - The Enum datatype, will be expressed by the *owl:oneOf* property composed of the different *enum* attribute values.
   - Temporal datatypes will be expressed by one of the many temporal XML schema datatypes.
   - Binary datatypes will be expressed by the *hexBinary* XML schema datatype. The *minLength* facet value is 0 if the attribute value is null, 1 otherwise.
2) The Primary key will be expressed by *OWL2 owl: Haskey* property on the class name representing the table containing this key and having for values the list of attributes participating in the primary key.
3) Each unique key (name) will be expressed as a subclass (of the class containing the unique key) containing the *owl: Haskey* property having for value the list of attributes participating in the unique key.
4) Foreign keys will be expressed by *owl:Restriction* property on the name of each attribute participating in the key (*owl:onProperty*) towards the referenced attribute (*owl:someValuesFrom*).

IV. EXAMPLE

This section provides an example on how to represent the schema of existing databases using Relational.OWL2E. Firstly, we will present the relational schema to describe: it is a MYSQL relational database ‘Breeding’ wich contains three tables ‘species’, ‘Race’ and ‘Animal’ then we will give some extracts from the generated OWL2 ontology.

A. Relational schema to describe

Create database Breeding ;
Create table Species (
id smallint(6) not null auto_increment,
latin_name varchar(40) not null,
primary key(id),
unique key latin_name (latin_name));
Create table Race (
id smallint(6) not null auto_increment,
species_id smallint(6),
primary key(id),
constraint fk_race_espece_id foreign key(species_id) references Breeding(id));
Create table Animal(
id smallint(6) not null auto_increment,
sex enum('male', 'female'),
Birth_date datetime not null,

```
name varchar(30),
species_id smallint(6) not null,
race_id smallint(6),
primary key(id),
constraint fk_species_id foreign key (species_id)
references Species(id),
constraint fk_race_id foreign key(race_id) references
Race(id));
```

B. A few Relationnal.OWL2E ontology extracts

Database Description

```
<owl:Class rdf:ID="Breeding">
<rdf:type rdf:resource="#Database">
<hasTable rdf:resource="#Species" />
<hasTable rdf:resource="#Race" />
<hasTable rdf:resource="#Animal" />
</owl:Class>
```

Table Description

```
<!—Species table description-->
<owl:Class rdf:ID="Species">
<rdf:type rdf:resource="#Table" />
<hasColumn rdf:resource="#Species.id" />
<hasColumn rdf:resource="#Species.latin_name" />
<hassynonym>Type</hassynonym>
<hassynonym>category</hassynonym>
<hasPrimaryKey>
<PrimaryKey>
<hasColumn rdf:resource="#Species.id" />
</PrimaryKey>
</hasPrimaryKey>
<hasUniqueKey>
<UniqueKey>
<hasColumn rdf:resource="#latin_name" />
</UniqueKey>
</hasUniqueKey>
</owl:Class>

<!—Race table description-->
<owl:Class rdf:ID="Race">
        <rdf:type rdf:resource="#Table" />
        <hasColumn rdf:resource="#Race.id" />
        <hasColumn rdf:resource="#Race.name" />
       <hasColumn rdf:resource="#Race.species_id" />
       <hasColumn rdf:resource="#Race.description" />
        <hasPrimaryKey>
        <PrimaryKey>
        <hasColumn rdf:resource="#Race.id" />
        </PrimaryKey>
        </hasPrimaryKey>
        <hasforeignkey rdf:about="#Species">
        <Column rdf:about="#Race.species_id">
        <references rdf:resource="#Species.id" />
        </Column>
        </hasReferenceTo>
        </owl:Class>
```

Attribute Description

```
< !-- Espece table id Attribute -->
<owl:DatatypeProperty rdf:ID="Species.id">
<rdf:type rdf:ressource="#Column" />
<rdfs:domain rdf:ressource="#Animal" />
<rdfs:range
rdf:ressource="http://www.w3.org/2001/XMLSchema
#short" />
</owl:DatatypeProperty>
<!-- Species table latin_name Attribute -->
<owl:DatatypeProperty rdf:ID="Species.latin_name">
<rdf:type rdf:ressource="#Column" />
<rdfs:domain rdf:ressource="#Species" />
<rdfs:range>
<rdfs:Datatype>
<owl:onDatatype
rdf:ressource="http://www.w3.org/2001/XMLSchema
#string" />
<owl:withRestrictions rdf:parseType="Collection">
<rdf:Description>
<xsd:minLength
rdf:datatype="http://www.w3.org/2001/XMLSchema
#integer">1</xsd:minlength>
</rdf:Description>
<rdf:Description>
<xsd:maxLength
rdf:datatype="http://www.w3.org/2001/XMLSchema
#integer">40</xsd:maxLength>
</rdf:Description>
</owl:withRestrictions>
</rdfs:Datatype>
</rdfs:range>
</owl:DatatypeProperty>
```

Primary key and unique key

```
<!-- Species table Primary key -->
<owl:Class rdf:about="Species">
<owl:hasKey rdf:parseType="Collection">
<owl:DatatypeProperty rdf:resource="#Species.id" />
</owl:hasKey>
</owl:Class>
<!-- Species table unique key -->
<owl:Class rdf:about="Species">
<owl:equivalentClass>
<owl:Class>
<owl:hasKey rdf:parseType="Collection">
<owl:DatatypeProperty
rdf:resource="#Species.latin_name" />
</owl:hasKey>
</owl:Class>
</owl:equivalentClass>
</owl:Class>
```

Foreign Key

```
<!-- Race table Foreign Key -->
<owl:Class rdf:about="Race">
<owl:equivalentClass>
<owl:Restriction>
<owl:onProperty rdf:ressource="#Race.Species_id" />
<owl:someValuesFrom rdf:ressource="#Species.id" />
</owl:Restriction>
</owl:equivalentClass>
</owl:Class>
```

V. Conclusion

The majority of the data on the web resides in Relational databases, the success of the Semantic Web hinges on offering efficient ways of integrating relational databases into the semantic web. This requires a prior description of them. In this work, we have presented our Relational.OWL2E new approach which generates the correspondent OWL2 ontology to a relational database schema. We tried to find the best OWL2 constructors to best express the relational concepts semantic.

The provided ontology can be improved to deal with other specific concepts and semantic properties. Our algorithm can be easily implemented for database management systems other than MySQL.

References

[1] G. Yang, J. Feng, "Database Semantic Interoperability based on Information Flow Theory and Formal Concept Analysis", International journal of information technology and computer science, PP.33-42, 2012, DOI: 10.5815/ijitcs.2012.07.05.

[2] C. Batini, M. Lenzerini and S.B. Navathe, "A Comparative Analysis of Methodologies for Database Schema Integration", In ACM Computing Surveys, Vol.18, No.4, 1986.

[3] V. Jain , M. Singh, "Ontology Based Information Retrieval in Semantic Web: A Survey", International journal of information technology and computer science,PP.62-69, 2013, DOI: 10.5815/ijitcs.2013.10.06

[4] N.S. Ougouti, H. Belbachir, Y. Amghar and A.N. Benharkat, "Architecture Of MedPeer : A New P2P-based System for Integration of Heterogeneous Data Sources", Proceedings of the International Conference on Knowledge Management and Information Sharing (KMIS), Paris,pp. 351-354, 2011.

[5] C. Golbreich and K. Wallace, "OWL2 Web Ontology Language New Features and Rationale", W3C Recommendation, 2009.

[6] C. Perez de Laborda and S. Conrad, "Relational.OWL – A Data and Schema Representation Format Based on OWL", Second Asia-Pacific Conference on Conceptual Modelling (APCCM), Newcastle, volume 43 of CRPIT, pp. 89–96. 2005.

[7] D. Dejing, P. LePendu, K. Shiwoong and Q. Peishen, "Integrating Databases into the Semantic Web through an Ontology-Based Framework", Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06), Washington, 2006.

[8] Intellidimension Company, "RDF Gateway", http://www.intellidimension.com, 2000.

[9] T.D.T. Nguyen, "A Dl-Based Approch To Integrate Relational Data Sources Into The Semantic Web", PHD thesis, Sophia Antipolis university, France 2008.

[10] N. Cullot, R. Ghawi and K. Yetongnon, "DB2OWL: A Tool for Automatic Database to Ontology Mapping", Proc. of 15th Italian Symposium on Advanced Database Systems (SEBD 2007), Torre Canne, pp. 491-494, 2007.

[11] J. Barrasa, O. Corcho, and A. Gomez-Perez, "R2O, an Extensible and Semantically Based Database-to-Ontology Mapping Language", Second Workshop on Semantic Web and Databases, 2004.

[12] J.F. Sequeda, S.H. Tirmizi, O. Corcho and D.P. Miranker, "Survey of Directly Mapping Sql Databases to the Semantic Web", Knowledge Eng. Review, 2012.

[13] M. Arenas, E. Prud'hommeaux, J. Sequeda, "Direct Mapping of Relational Data to RDF", W3C Working, 2011.

[14] I. Astrova, N. Korda, and A. Kalja, "Rule-Based Transformation of SQL Relational Databases to OWL Ontologies" , 2nd International Conference on Metadata & Semantic Research, 2007.

[15] F. Barbancon and D. P. Miranker, "SPHINX: Schema integration by example", Journal of Intelligent Information Systems , 29 (2), 2007.

Authors' Profiles

Naïma S. Ougouti is born in 1971. She Holds an Engineer degree in the field of computer science in 1994 at Es-senia university of Oran (Algeria), then a Magister degree in 2004 at the university of sciences and technology of Oran –Mohamed Boudiaf (USTO-MB). She worked as Engineer of computer science in a big Algerian petroleum company from 1996 to 2005 then since 2006, she is an Assistant Professor at (USTO-MB) university. Her field of teaching concerns operating systems, networks and databases. She is also a Member of LSSD laboratory and an active participant in some research projects. Her current research interests include semantic web, information retrieval and interoperability.

Hafida Belbachir is born in 1955. She is Professor of computer Sciences at the University of Sciences and Technology of Oran–Mohamed Boudiaf (USTO-MB) in Algeria. She received her PHD in Computer Science at University of Oran in 1990. She heads the Database System Group in the LSSD Laboratory. Her research interests include Advanced DataBases, DataMining and Data Grid. She is author of more than 80 papers in reviews and proceedings.

Youssef Amghar is born in 1956. He is Professor of management information systems at the Scientific and Technical University of Lyon where he is the head of Computer Science Department. He holds a PhD in Computer Science from the same university in 1989 following by an HDR in 1997.

His field of teaching concern project management, databases and development processes. His is an active member of laboratory of information system of INSA de Lyon. His current research interests include information retrieval, interoperability of applications and legal documents. He is author of more than 100 papers related to these research activities and managed some projects about decisions support. Actually, he is responsible of research team working on the domain of service oriented computing.

How to cite this paper: Naïma S. Ougouti, Hafida Belbachir, Youssef Amghar,"A New OWL2 Based Approach for Relational Database Description", International Journal of Information Technology and Computer Science(IJITCS), vol.7, no.1, pp.48-53, 2015. DOI: 10.5815/ijitcs.2015.01.06