

Relational.OWL2E : Une nouvelle approche de représentation du schéma d'une base de données relationnelle basée sur OWL2

Naïma Souâd Ougouti*, Hafida Belbachir*, Dolière Francis Some*,
Ismael Abraham Ouattara*

* Laboratoire LSSD, USTO-MB BP 1505 El M'Naouer,
Oran, Algeria

{ s_ougouti, h_belbach, some.frncs, ismaelouattara03 }@yahoo.fr

Résumé. Un des domaines de recherche qui intéressent la communauté scientifique de nos jours est le problème de la médiation dans les systèmes pair-à-pair (P2P). Dans ce contexte nous avons proposé un nouveau système de gestion de données hétérogènes et distribuées dans un environnement P2P nommé MedPeer. Parmi les fonctions de ce système, nous nous intéressons dans cet article à la description des bases de données relationnelles en utilisant des ontologies. Nous proposons donc Relational.OWL2E, une nouvelle approche qui génère à partir du schéma relationnel une ontologie basée sur le langage OWL dans sa deuxième version (OWL2).

1 Introduction

On assiste depuis quelques années à l'émergence de nouvelles applications qui ont besoin de partager des informations entre différents systèmes. C'est le cas du e-gouvernement, du e-Learning, du e-commerce, de la bioinformatique ou encore des bibliothèques électroniques. Or, dans ce contexte, les systèmes d'informations, conçus et développés par des organisations différentes, constituent généralement des sources de données hétérogènes et autonomes.

Ainsi, l'interopérabilité est devenue une nécessité pour répondre au besoin d'échange d'informations entre systèmes d'informations hétérogènes. Elle traduit la capacité d'un système d'informations à collaborer avec d'autres systèmes de nature parfois très différente. Elle a pour objectif de développer des architectures et des outils pour le partage, l'échange et le contrôle des données.

Dans ce contexte, nous avons présenté un nouveau système d'intégration des données dans un environnement P2P nommé MedPeer (Ougouti et al., 2011). Ce système repose sur une architecture Super-pair s'appuyant sur un regroupement des pairs selon le type de média (Textes, Images, Bases de données relationnelles, semi-structurées,...). Cette architecture combine l'approche centralisée et celle non structurée prenant ainsi les avantages de la recherche centralisée, de l'autonomie, de la répartition des charges et de la robustesse pour une recherche distribuée.

Chaque super-pair gère les pairs traitant le même type de média qu'il est en charge de représenter, il est choisi en fonction de ses capacités en terme de capacités de calcul et de

bande passante. Il doit en outre disposer de toutes les informations nécessaires pour pouvoir orienter les requêtes qui lui arrivent vers les pairs pertinents. Les super-pairs forment entre eux un réseau pair-à-pair pur. Lorsque les pairs ont des schémas différents à gérer, une médiation sémantique est nécessaire. Dans cet article, nous nous intéressons à ce dernier problème puisque nous présentons un nouveau format de représentation pour les schémas relationnels basé sur le langage d'ontologies Web OWL dans sa deuxième version nommé Relationnel.OWL2E. En exploitant les différentes opportunités fournies par OWL2 (Golbreich C. et Wallace K, 2009) et notre ontologie, nous sommes capables aujourd'hui de décrire et de partager n'importe quel schéma d'une base de données relationnelle.

Cet article est organisé comme suit :

Dans la section 2, nous présentons un état de l'art des principales approches de description des bases de données relationnelles. En section 3, nous introduisons Relationnel.OWL2E, notre ontologie OWL2 pour la représentation d'un schéma relationnel. Dans la section 4, nous illustrons notre approche par un exemple. Enfin nous terminons par une conclusion.

2 Etat de l'art

En voulant tirer profit des bénéfices qu'apporte le web sémantique, plusieurs travaux dont le but est le passage d'une base de données relationnelle à un format plus récent (XML/RDF/OWL) ont vu le jour. Nous avons choisi de présenter quatre approches (Perez de Laborda et al. , 2005) (Dejing D. et al., 2006) (Intellidimension Company, 2000) (Nguyen T.D.T, 2008), d'autres méthodes plus récentes se trouvent dans (Sequeda J.F., 2012) (Arenas M. et al., 2011) (Cullot, N et al., 2007).

Relational.OWL (Perez de Laborda et al. , 2005) permet de traduire presque tous les concepts du modèle relationnel en ontologies OWL, du schéma relationnel aux données en passant par les contraintes d'intégrité. Ce système définit quatre classes et un ensemble de propriétés qui permettent de relier entre elles.

Dans le tableau1 sont répertoriées les classes prédéfinies et dans le tableau 2 se trouvent les différentes propriétés.

Les préfixes rdf, rdfs, dbs, xsd et owl représentent des espaces de noms utilisés dans l'ontologie Relational.OWL

rdf :ID	rdfs :subClassOf	rdfs :comment
dbs : Database	rdf :Bag	Classe des bases de données
dbs :Table	rdf :Seq	Classe des tables de la base
dbs :Column	rdf :Ressource	Classe des colonnes
dbs :PrimaryKey	rdf:Bag	Clé primaire d'une table

TAB.1-Classes de Relational.OWL Classe des bases de données

rdf :ID	rdfs :domain	rdfs :range	rdfs :comment
dbs :hasTable	dbs:Database	dbs:Table	Ensemble de tables
dbs :hasColumn	dbs:Table	dbs:Column	Ensemble de colonnes
dbs:isIdentifiedBy	dbs:Table	dbs:PrimaryKey	Clés primaires des tables
dbs:references	dbs:Column	dbs:Column	Clés étrangères
dbs:length	dbs:Column	xsd:nonNegativeInteger	Longueur d'un champ.
dbs:scale	dbs:Column	xsd:nonNegativeInteger	Partie décimale

TAB.2- Propriétés de Relational.OWL

OntoGrate (Dejing D. et al., 2006) est un système d'intégration de bases de données relationnelles dans un environnement P2P (Pair-à-Pair). Pour représenter les schémas relationnels en ontologies OWL, les auteurs ont étendu l'expressivité des langages d'ontologies web. Ils ont ainsi introduit un nouveau langage, Web-PDDL, extension de PDDL (*Planning Domain Definition Language*) basé sur la logique des prédicats du premier ordre. Dans un premier temps, les concepts de la base de données sont traduits en utilisant le langage Web-PDDL. Une fois l'ontologie générée, le système dispose d'un adaptateur de syntaxe, PDDOWL, qui traduit ainsi la première ontologie Web-PDDL en ontologie OWL. Dans l'ontologie finale générée, une table est transformée en une classe, sous classe de la classe sql:Relation (Définie dans le système OntoGrate, comme étant la classe représentant les tables), un attribut est transformé en propriété OWL, une contrainte est vue comme un axiome (règle) et une clé primaire comme une contrainte fonctionnelle OWL (owl:FunctionalProperty).

RDF Gateway (Intellidimension Company, 2000) est un système qui permet de traduire le schéma relationnel d'une base de données en une ontologie RDFS ou OWL, via le paramètre *schema_type*, qui spécifie la sortie par défaut de l'ontologie.

Le SQL Data service est un module du système RDF Gateway qui interroge la base de données et en extrait le schéma relationnel qu'il traduit en ontologies RDFS ou OWL. Dans ce système une table est traduite en classe, un attribut en une propriété *rdfs:property* pour une sortie RDFS ou *owl:DatatypeProperty* pour une sortie OWL, une clé étrangère en une propriété *rdfs:property* ou *owl:ObjectProperty* et enfin les types de données des attributs sont traduits en type de données du Schéma XML.

OWL_K (K pour *Key* ou clé) (Nguyen T.D.T, 2008) est une extension du langage OWL pour la gestion des contraintes d'identification. Les contraintes d'identification sont l'équivalent des clés primaires du modèle relationnel. Ce travail a été motivé par les difficultés du dialecte OWL DL à capturer toute la sémantique des contraintes d'identification. Le vocabulaire par défaut de OWL a été étendu afin de prendre en compte les contraintes d'identification.

Le système propose :

- La classe *ICAssertion* qui représente la contrainte d'identification.
- La propriété *onClass* qui est la classe (table) sur laquelle porte la contrainte d'identification
- La propriété *byProperty* qui représente une propriété (attribut) participant à la contrainte d'identification.

Le langage de logique descriptive par défaut de OWL a aussi été étendu pour prendre en compte la sémantique des nouveaux concepts introduits.

Dans ce système, les types de données sont traduits en utilisant XML Schéma et les clés étrangères sont traduites en utilisant les contraintes de cardinalités (*owl:minCardinality*, *owl:cardinality*, *owl:maxCardinality*).

3 Relational.OWL2E

La solution que nous proposons ici est une extension du système Relational.OWL proposé en (Perez de Laborda et al. , 2005). Notre choix s'est porté sur cette approche à cause de sa particularité à traduire presque tous les concepts du modèle relationnel en ontologies OWL.

Notre apport majeur se situe au niveau de la sémantique que nous avons ajoutée aux différents concepts des bases de données relationnelles, en traduisant entre autres, les attributs par des types de données riches du schéma XML, les clés primaires, les clés uniques, les clés étrangères, et en tenant compte des contraintes NULL et NOT NULL du modèle relationnel.

Nous avons appelé cette ontologie Relational.OWL2E parce qu'elle est basée sur OWL2 et sur Relational.OWL que nous avons étendu (E).

Nous obtenons les informations sur le contenu de la base de données à partir de son dictionnaire de données (catalogue) et nous donnons dans ce qui suit la représentation des tables, attributs (colonnes), types de données avec éventuellement les restrictions nécessaires, les clés primaires, les clés uniques et les clés étrangères.

Nous avons donc défini 5 classes et 7 relations (propriétés) entre elles, elles sont résumées dans les deux tableaux suivants :

Classes	Commentaires
Database	Classe représentant une base de données
Table	Classe représentant une table
Column	Classe représentant une colonne
PrimaryKey	Classe représentant une clé primaire
UniqueKey	Classe représentant une clé unique

TAB3.- Classes dans Relational.OWL2E

Propriétés	rdfs :domain	rdfs :range	Commentaires
has	owl :Thing	owl :Thing	Une chose possède une autre chose.
hasTable	Database	Table	Une table fait partie d'une base de données
hasColumn	Table PrimaryKey UniqueKey	Column	Une colonne fait partie d'une table
hasPrimaryKey	Table	PrimaryKey	Une clé primaire identifie une relation
hasUniqueKey	Table	UniqueKey	Une table peut avoir des contraintes d'unicité sur certains attributs
hasForeignKey	Table	Table	Une table référence une autre table dans une relation de clé étrangère.
references	Column	Column	Une colonne référence une autre colonne dans une clé étrangère.

TAB.4-Propriétés dans Relational.OWL2E

3.1 Sérialisation de l'ontologie Relational.OWL2E.

Dans ce qui suit nous donnons quelques extraits de sérialisation, dans la syntaxe RDF/XML de l'ontologie Relational.OWL2E dont les classes et les propriétés ont été décrites ci-dessus.

Définition d'une classe

```
<owl:Class rdf:ID="Table">
  <rdfs:subClassOf rdf:about="http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag" />
  <rdfs:label xml:lang="fr">Table</rdfs:label>
  <rdfs:label xml:lang="en">Table</rdfs:label>
  <rdfs:comment xml:lang="fr">Classe des tables d'une base de données.</rdfs:comment>
  <rdfs:comment xml:lang="en">The class of database tables.</rdfs:comment>
</owl:Class>
```

Définition d'une propriété

```
<owl:ObjectProperty rdf:ID="hasTable">
  <rdfs:subPropertyOf rdf:resource="#has" />
  <rdfs:domain rdf:resource="#Database"/>
  <rdfs:range rdf:resource="#Table" />
```

```

    <rdfs:label xml:lang="fr">aPourTable</rdfs:label>
    <rdfs:label xml:lang="en">hasTable</rdfs:label>
    <rdfs:comment xml:lang="fr">Une base de données
    contient un ensemble de tables </rdfs:comment>
    <rdfs:comment xml:lang="en">A Database has a set of
    Tables.</rdfs:comment>
</owl:ObjectProperty>

```

3.2 Algorithme de passage d'une base de données relationnelle à une ontologie OWL 2 : cas de MySQL

3.2.1 Première partie

On commence par extraire les tables d'une base de données

- Le nom de la base de données représente une classe, de type *Database* de *Relational.OWL2E*
- Le nom de chaque table sera exprimé comme valeur de la propriété *hasTable* de *Relational.OWL2E*

Pour chaque table, on extrait la liste des attributs, la clé primaire, les clés uniques et étrangères.

- Le nom de chaque table est une classe de type *Table* de *Relational.OWL2E*
- Le nom de chaque attribut sera exprimé comme valeur de la propriété *hasColumn*
- La clé primaire sera exprimée par la propriété *hasPrimaryKey* sur la classe *PrimaryKey* (représentant la clé primaire) et contenant la liste des attributs participant à la clé, chaque attribut étant exprimé comme valeur de la propriété *hasColumn*
- La clé unique sera exprimée similairement à la clé primaire, mais avec les propriétés *hasUniqueKey* vers la classe *UniqueKey* contenant la liste des attributs participants à la clé, chaque attribut étant exprimé comme valeur de la propriété *hasColumn*
- La clé étrangère sera exprimée par la propriété *hasForeignKey*. La valeur de cette propriété sera la table référencée par la clé étrangère. Chaque attribut de la clé étrangère sera exprimé comme instance de la classe *Column* et liée à la colonne référencée par la propriété *references* ayant pour valeur l'attribut référencé.

Le résultat de cette première partie de l'algorithme sera une ontologie reprenant les termes du schéma d'une base de données relationnelle dans les termes du schéma de l'ontologie que nous avons défini.

3.2.2 Deuxième Partie

1) Chaque attribut sera exprimé comme une propriété de type de données, dont le domaine (*rdfs:domain*) est le nom de la classe représentant la table contenant l'attribut et l'image (*rdfs:range*) le type de données de l'attribut traduit en un type de données du schéma XML de la façon suivante :

- Les types de données entier seront exprimés par le type *integer* du schéma XML, avec éventuellement des restrictions sur les intervalles de valeurs du type, grâce aux facettes *maxInclusive*, *maxExclusive*, *minInclusive*, *minExclusive* du schéma XML.

- Les types de données numériques à virgules (décimaux), nous les exprimons par le type *decimal* du schéma XML avec éventuellement des restrictions grâce aux facettes *totalDigits* (nombre total de caractères du type) et *fractionDigits* (nombre de chiffres après la virgule).
 - Les types de données textes seront exprimés par le type *string* du schéma XML. Nous utilisons les facettes *minLength* et *maxLength* pour exprimer le nombre de caractère minimal et maximal autorisé dans les chaînes de ce type.
 - Pour la valeur de la facette *minLength*, si l'attribut accepte les valeurs nulles, alors, *minLength* vaudra 0, sinon 1
 - Pour le type set, on le traduit par un type string, dont la valeur de la facette *maxLength* est à extraire du catalogue MySQL.
 - Pour le type de données *enum*, on l'exprimera par la propriété *owl:oneOf* composée des différentes valeurs prises par l'attribut *enum*.
 - Les types temporels seront exprimés par l'un des nombreux types temporels du schéma XML.
 - Les types de données binaires, sont similaires aux types de données textes, à la seule différence qu'ils sont exprimés par le type *hexBinary* du schéma XML. Comme dans le cas des types textes, la valeur de la facette *minLength* est 0 si les valeurs de l'attribut peuvent être nulles ou 1 sinon.
- 2) La clé primaire sera exprimée par la propriété *owl:hasKey* de OWL2 sur le nom de la classe représentant la table contenant cette clé et ayant pour valeurs la liste des attributs participant à la clé primaire.
 - 3) Chaque clé unique (son nom) sera exprimée comme une sous-classe (de la classe contenant la clé unique) contenant la propriété *owl:hasKey* sur une classe équivalente de la classe représentant la table contenant la clé unique et ayant pour valeurs la liste des attributs participant à la clé unique
 - 4) Les clés étrangères seront exprimées par des restrictions de propriétés (*owl:Restriction*) sur le nom de chaque attribut participant à la clé (*owl:onProperty*) vers l'attribut référencé (*owl:someValuesFrom*).

4 Exemple d'application

Dans cette section, nous donnons quelques extraits de l'ontologie OWL qui décrit la base de données relationnelle *Elevage* sous MySQL. Elle est constituée de trois tables *Espec*, *Race* et *Animal*.

4.1 Schéma relationnel à décrire

```
CREATE DATABASE Elevage ;
```

```
CREATE TABLE Espec (
  id smallint(6) not null auto_increment,
  nom_latin varchar(40) not null,
  primary key(id),
  unique key nom_latin (nom_latin));
```

```

CREATE TABLE Race (
  id smallint(6) not null auto_increment,
  espece_id smallint(6),
  primary key(id),
  constraint fk_race_espece_id foreign key(espece_id) references Espece(id));

```

```

CREATE TABLE Animal(
  id smallint(6) not null auto_increment,
  sexe enum('male', 'femelle'),
  date_naissance datetime not null,
  nom varchar(30),
  espece_id smallint(6) not null,
  race_id smallint(6),
  primary key(id),
  constraint fk_espece_id foreign key (espece_id) references Espece(id),
  constraint fk_race_id foreign key(race_id) references Race(id));

```

4.2 Quelques extraits de l'ontologie Relationnal.OWL2E générée

Description de la base de données

```

<owl:Class rdf:ID="Elevage">
  <rdf:type rdf:resource="#Database">
  <hasTable rdf:resource="#Espece" />
  <hasTable rdf:resource="#Race" />
  <hasTable rdf:resource="#Animal" />
</owl:Class>

```

Description d'une table

```

<owl:Class rdf:ID="Espece">
  <rdf:type rdf:resource="#Table" />
  <hasColumn rdf:resource="#Espece.id" />
  <hasColumn rdf:resource="#Espece.nom_latin" />
  <hasPrimaryKey>
    <PrimaryKey>
      <hasColumn rdf:resource="#Espece.id" />
    </PrimaryKey>
  </hasPrimaryKey>
  <hasUniqueKey>
    <UniqueKey>
      <hasColumn rdf:resource="#nom_latin" />
    </UniqueKey>
  </hasUniqueKey>
</owl:Class>

```

Description d'un attribut

<!--Attribut id de la table Espece--!>

```
<owl:DatatypeProperty rdf:ID="Espece.id">
    <rdf:type rdf:resource="#Column" />
    <rdfs:domain rdf:resource="#Animal" />
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#short" />
```

```
</owl:DatatypeProperty>
```

<!--Attribut nom_latin de la table Espece-->

```
<owl:DatatypeProperty rdf:ID="Espece.nom_latin">
<rdf:type rdf:resource="#Column" />
<rdfs:domain rdf:resource="#Espece" />
<rdfs:range>
<rdfs:Datatype>
<owl:onDatatype
rdf:resource="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>
```

Clé primaire et clé unique

<!-- Clé primaire de la table Espece -->

```
<owl:Class rdf:about="Espece">
    <owl:hasKey rdf:parseType="Collection">
        <owl:DatatypeProperty rdf:resource="#Espece.id" />
    </owl:hasKey>
```

```
</owl:Class>
```

<!-- Clé unique de la table Espece -->

```
<owl:Class rdf:about="Espece">
    <owl:equivalentClass>
        <owl:Class>
```

```

        <owl:hasKey rdf:parseType="Collection">
        <owl:DatatypeProperty
rdf:resource="#Espece.nom_latin" />
        </owl:hasKey>
        </owl:Class>
    </owl:equivalentClass>
</owl:Class>

```

Clé étrangère

```

<!-- Clés étrangères de la table Race -->
<owl:Class rdf:about="Race">
    <owl:equivalentClass>
    <owl:Restriction>
    <owl:onProperty rdf:resource="#Race.espece_id" />
    <owl:someValuesFrom rdf:resource="#Race.id" />
    </owl:Restriction>
    </owl:equivalentClass>
</owl:Class>

```

5 Conclusion

L'intégration des bases de données relationnelles dans le web sémantique passe par une description préalable de celles-ci.

Dans ce travail, nous avons introduit notre nouvelle approche qui a cet objectif en utilisant une ontologie. Nous avons pour cela détaillé les différentes étapes de l'algorithme qui mènent d'un schéma relationnel vers l'ontologie Relationnal.OWL2E écrite avec le langage OWL2, tout en respectant toute la sémantique contenue dans la structure du schéma.

L'algorithme fourni peut être facilement implémenté pour des systèmes de gestion de bases de données relationnelles autres que MySQL.

L'ontologie fournie, peut être améliorée et corrigée pour prendre en charge d'autres concepts spécifiques.

Références

- Arenas M., Prud'hommeaux E. and Sequeda J, (2011), Direct mapping of relational data to RDF. W3C Working
- Cullot, N., Ghawi, R. and Yetongnon, K. (2007), DB2OWL: A Tool for Automatic Database to Ontology Mapping. In: Proc. of 15th Italian Symposium on Advanced Database Systems (SEBD 2007), Torre Canne, 491-494.
- Dejing D., LePendu P., Shiwoong K., and Peishen Q., (2006), Integrating Databases into the Semantic Web through an Ontology-Based Framework. In ICDEW '06: Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06), Washington.

- Golbreich C. et Wallace K. (2009), OWL2 Web Ontology Language New Features and Rationale. W3C Recommendation.
- Intellidimension Company. RDF Gateway, (2000). Available at <http://www.intellidimension.com>.
- Nguyen T.D.T. ,(2008), A DI-Based Approach To Integrate Relational Data Sources Into The Semantic Web. Thèse de doctorat, Université de Nice-Sophia Antipolis, France.
- Ougouti, N.S., Belbachir, H., Amghar and Y., Benharkat, A.N., (2011), Architecture Of MedPeer : A New P2P-based System for Integration of Heterogeneous Data Sources. In: Proceedings of the International Conference on Knowledge Management and Information Sharing (KMIS 2011), Paris, 351-354.
- Perez de Laborda et C., Conrad S., (2005), Relational.OWL – A Data and Schema Representation Format Based on OWL. In: Second Asia-Pacific Conference on Conceptual Modelling (APCCM2005), Newcastle, volume 43 of CRPIT, 89–96.
- Sequeda J.F., Tirmizi S.H., Corcho O., and Miranker D.P. (2012), Survey of directly mapping sql databases to the semantic web. Knowledge Eng. Review.

Summary

Nowadays, scientific community is more and more interested by the mediation problem into Peer-to-Peer systems (P2P) and by migration of data sources into semantic web. 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 are interested in this article in relational databases description by using ontologies. We thus propose Relational.OWL2E, a new approach which generates starting from relational schema an OWL2 based ontology.