Université Abou Bekr Belkaid
Tlemcen Algérie
جامعة أبي بكر بلقايد
تلمسان الجزائر

**République Algérienne Démocratique et Populaire**
**Université Abou Bakr Belkaid– Tlemcen**
**Faculté des Sciences**
**Département d'Informatique**

**Mémoire de fin d'études**

**Pour l'obtention du diplôme de Master en Informatique**

***Option :*** *Système d'Information et de Connaissances (S.I.C)*

*Thème*

# Transformation d'une Base de données relationnelle en Linked Data

**Réalisées par :**

- **HASNAOUI Nabila**
- **DJEZIRI Salima**

*Présenté le 24 Juin 2014 devant le jury composé de MM.*

- Mr Belabed A. *(Président)*
- *Melle Berramdane Dj.* *(Encadreur)*
- *Mr* Chouiti S. *(Examinateur)*
- Mme Iles N. *(Examinateur)*

Année universitaire: 2013-2014

# Remerciements

***Nous tenons tout d'abord à remercier DIEU le tout puissant pour nous avoir donné la force de réaliser ce travail, et aussi nos parents pour leurs soutiens et encouragements durant nos années d'études.***

***Nous remercions profondément notre encadreur Melle BERRAMDANE Djamila pour son aide, ses encouragements et ses critiques constructives qui nous ont beaucoup aidés à apprécier ce travail.***

***Nous remercions également les membres du jury qui ont accepté de participer à la discussion de notre travail.***

***Nous exprimons aussi toute notre gratitude aux enseignants du département d'informatique ainsi que tous nos collègues et nos amis et à tous ce qui ont contribué de prés ou de loin à la réalisation de notre mémoire.***

*Nabila et Salima*

# dédicaces

***Nous dédicaçons ce modeste travail à:***

- ***A nos très chers parents.***
- ***A nos maris.***
- ***A nos frères et sœurs.***
- ***A nos belles familles.***
- ***A nos amies.***
- ***A nos collègues.***

*Nabila et Salima*

# Table des matières

# Liste des figures

# Liste des Tableaux

# *Introduction générale*

**Contexte :**

Le World Wide Web représente aujourd'hui une source d'information en constante évolution. L'accès au réseau Internet étant de plus en plus disponible, la quantité d'informations contenues dans le Web, différant par contenu, type, forme ou thématique, a connu une croissance exponentielle.

Face à cela, accéder rapidement et automatiquement à ces informations est devenu fondamental.

La dernière décennie a connu l'essor du Web Sémantique (ou Web 3.0, ou Web des données) en réponse à un besoin de représenter les contenus textuels du Web de façon sémantiquement structurée, afin de faciliter le traitement automatique et l'interrogation avec des moteurs de recherche sémantiques.

Cependant, les contenus non-structurés du Web des documents restent encore peu exploitables par les technologies du Web Sémantique. D'autre part, il est encore difficile d'envisager un passage à l'utilisation à large échelle du Web Sémantique de la part des utilisateurs, à cause de la grande quantité d'informations et domaines à traiter [1].

Un ensemble de principes et de technologies qui ne cessent de se développer, est connu sous le nom de données liées (Linked Data). Aujourd'hui, il est capable d'exploiter la philosophie et l'infrastructure du Web pour permettre le partage de données et leur réutilisation à grande échelle.

Afin de comprendre le concept et la valeur des données liées, il est important de considérer les mécanismes actuels d'échange et de réutilisation de données sur le Web.

Un facteur-clé dans la réutilisation des données est la façon dont elles sont structurées. Plus cette structure est régulière et bien définie, plus cela facilite la création d'outils pour traiter et réutiliser les données de manière fiable.

Les recherches récentes ont montrés que les technologies Web sémantique sont utiles au-delà du Web, en particulier si les données provenant de différentes sources doivent être échangées ou intégrées, en plus la majorité des données sur le Web en cours sont stockées dans des bases de données relationnelles.

Par conséquent un domaine de recherche récent est apparu à savoir Mapping des bases de données relationnelles.

Les bases de données sont actuellement au cœur du système d'information des entreprises. Avec l'avènement des technologies du Web sémantique, les bases de données relationnelles ont

montré certains lacunes à savoir manque de la notion de partageabilité, le raisonnement sur les données en fin la sémantique des données.

**Problématique :**

Quelles sont les approches les plus optimales pour la mise en œuvre des Linked Data, en tenant compte des différentes technologies proposées autour du web ainsi que des différentes approches proposées jusqu'aujourd'hui, et qui permettront la disponibilité pour rendre les données relationnelles disponibles pour les applications Web sémantiques?

**Organisation du mémoire**

Ce mémoire est organisé comme suit :

La première partie comporte deux chapitres :

Le premier introduit la notion du web sémantique, en détaillant une autre notion qui est Les Linked Data, à l'ensemble des principes, technologies qui le définissent, les concepts ainsi que l'évolution de Linked Data.

Le deuxième chapitre, concerne Les bases de données relationnelles, ou autrement dit, quelques notions sur les bases de données.

La seconde partie du mémoire est consacrée aux approches de transformation de base de données en données liées du web ainsi nous présentons la partie implémentation de notre projet.

Nous clôturons ce mémoire par une conclusion.

# Première partie

# Background

# *CHAPITRE I:*

# *LINKED DATA*

## I.1 INTRODUCTION :

Le World Wide Web connaît depuis des années un développement incessant en terme de quantité d'informations disponibles et d'utilisateurs ainsi que de contributeurs. Son succès est dû principalement à la simplicité de sa structure, qui a permis de développer, fournir, atteindre et utiliser aisément du nouveau contenu

En 2010, on compte que le Web contient plus que 2 milliards de pages. L'accès aux informations qui y sont contenues devient enfin de plus en plus compliqué. Ces informations sont en effet exprimées en langage naturel, et ne sont pas accessibles par une machine (machine-readable).

Face à cela, Tim Berners-Lee propose en 2000 une nouvelle architecture, pour "amener le Web à exploiter son vrai potentiel". Le Web Sémantique a pour objectif de donner une structure aux contenus sémantiques du Web, grâce à laquelle les machines peuvent accéder aux données plus facilement. Plus précisément : rendre les machines capables de comprendre la sémantique des données et documents fournis dans le Web [2].

## I.2 DEFINITION DU WEB SEMANTIQUE :

Le Web Sémantique est une vision du futur Web dans lequel l'information est donnée un sens explicite facilitant ainsi aux machines le traitement et l'intégration des informations sur le Web. Le Web Sémantique sera construit sur la capacité de XML de définir des schémas de balisage personnalisés et sur la flexibilité de l'approche RDF pour représenter les données [3].Son but est de rendre la connaissance "universelle".

Il a pour objectif de donner une structure aux contenus sémantiques du Web, grâce à laquelle les machines peuvent accéder aux données plus facilement. Plus précisément : rendre les machines capables de comprendre la sémantique des données et documents fournis dans le Web. Le Web Sémantique n'est en effet vu que comme une extension du Web actuel, dans lequel humains et machines peuvent collaborer, au-delà des capacités actuelles, en utilisant une information dont le contenu sémantique est bien défini. Ce processus permet aux machines de "comprendre" le sens des données utilisées [4].

## I.3 DEFINITION DE LINKED DATA:

Les Linked Data (en français. Données liées) est désigné sous diverses appellations soulignant ses différentes propriétés. Il est souvent considéré comme la nouvelle génération de l'internet (appelé pour cela Web 3.0) sont un ensemble de données provenant de différentes

sources, homogènes ou hétérogènes et qui sont liées entre elles de manière typique [19]. Il s'agit d'un ensemble de technologies visant d'une part à publier en accès libre les données contenues dans des bases de données et cela dans des formats compréhensibles aisément et exploitables par des systèmes tiers, et d'autre part à relier les données avec d'autres sources publiées en ligne pour finalement constituer un réseau mondial de données accessibles ouvertement à tous (on parle alors de Linked Open Data). Ainsi, les Informations publiées sur internet, présentées jusque-là sous forme de documents compréhensibles par les humains, sont transformées en données ayant du sens pour les machines (on parle dans ce cas du Web sémantique) [17].

Le lancement de l'initiative *«Linking Data»* par le W3C avait pour objectifs :

- de promouvoir une vision du Web comme une base de données globale,
- et de relier les données sur le Web de la même façon que l'hypertexte permet de relier des documents (les pages Web)[19].

## I.4 PRINCIPES DE MISE EN OEUVRE :

Les termes *donnés liées* se réfèrent à un ensemble de bonnes pratiques à mettre en œuvre pour publier et lier des données structurées sur le Web. Ces pratiques ont été introduites par Tim Berners-Lee, concepteur du web et président du W3C dans *Linked Data* [2], sa note sur l'architecture du Web, Il publia en 2006 les *4 principes fondamentaux principes du Web de données liées*. Les voici :

- Nommer les éléments avec des URI ;
- Utiliser des URI HTTP, pour que l'on puisse rechercher/consulter ces noms ;
- Fournir des informations nécessaires sous forme de standards (RDF, SPARQL) lors d'une recherche d'URI ;
- Inclure des liens vers d'autres URI qui permettent de découvrir d'autres éléments. [5]

L'idée consiste à appliquer l'architecture du *World Wide Web* pour partage des données structurées à une échelle globale. Afin de comprendre ces principes, il est au préalable nécessaire de comprendre l'architecture du document Web classique.

Le Web est construit sur un ensemble de standards simples :

- Des URI (*Uniform Resource Identifiers*, identifiants de ressource uniformes) comme mécanisme d'identification unique et global ;
- HTTP (*HyperText Transfer Protocol*, protocole de transfert hypertexte), le mécanisme d'accès universel ;

- HTML (*HyperText Markup Language*, langage de balisage hypertexte), le format de contenu largement utilisé.

De plus, il s'appuie sur le principe de liens existant entre des documents pouvant résider sur des serveurs différents.

### *I.4.1 URI :*

URI (Uniform Resource Identifier - identifiant uniforme de ressource). L'URI est employé pour désigner le nom ou l'adresse (ou les deux) d'une ressource dans le Web de données sans distinction d'une ressource physique (donc sa représentation est récupérable via l'Internet telle qu'une page web, un service localisé sur un serveur…) ou d'une ressource abstraite (un livre particulier, une idée…).**URL** (Uniform Resource Locator), comme l'**URI**, est une chaîne courte des caractères mais il n'est employé que pour référer des ressources (physiques) par leur localisation[2]. Notons aussi que quelque chose qui peut être identifié avec un URI peut être décrit, ainsi le Web sémantique peut raisonner au sujet des personnes, des endroits, des idées… Les URIs sont utilisés pour identifier (nommer) des ressources, mais sans procédé pour les récupérer.

D'une certaine manière l'ensemble des URL est inclus dans l'ensemble des URI.

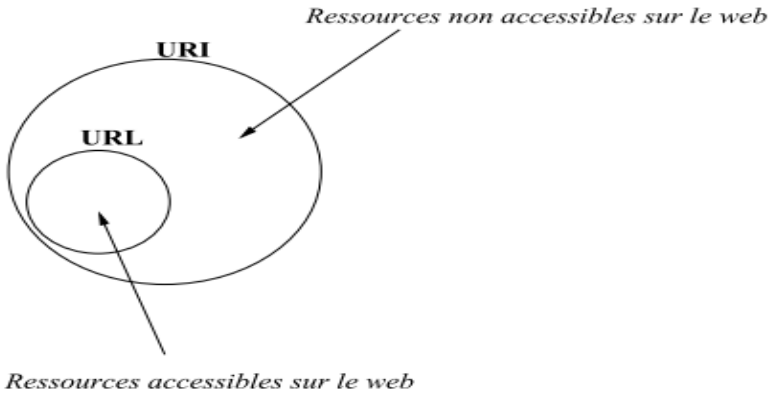


**Figure 01** : Ensembles des URI et des URL

Quelques exemples d'URI :

_ http://www.inria.fr/acacia/index.htm (pour une page web)

_ http://www.inria.fr/acacia/OntologyMatching.pdf (pour un article)

_ rstp://www.video.com/france.rm (pour une vidéo)

_ urn:issn:2242-4157 (pour un livre)

_ tel:+33-497-15-53-17 (pour un numéro de téléphone).

## *I.4.2 RDF :*

RDF (Resource Description Framework) n'est pas à proprement parler un langage. Il s'agit plutôt d'un modèle de données pour décrire des ressources sur le web. On entend par ressource toute entité que l'on veut décrire sur le web mais qui n'est pas nécessairement accessible sur le web.

RDF nous permet d'annoter sémantiquement les ressources, il propose un modèle de données simple fournissant un langage de représentation des propriétés et des relations des ressources du Web [13]

✓Représentation d'informations sur les ressources du Web.

✓Information destinée aux applications (pas aux humains) d'extraction d'information, ou aux services web.

✓Structure de graphe orienté : on décrit des arcs [6].

Basé sur le méta- langage eXtensible Markup Language (XML), RDF est la pierre angulaire du web sémantique. RDF établi des relations en utilisant le concept de triplet [9].

RDF est un modèle de graphes censé représenter les diverses ressources du Web afin d'en permettre le traitement automatique. Ce schéma RDF est représenté par un ensemble d'énoncés (statements). Chaque énoncé est un triplet <**S**, **P**, **O**> d'où **S** est le sujet, **P** est le prédicat, **O** est l'objet [ 7].

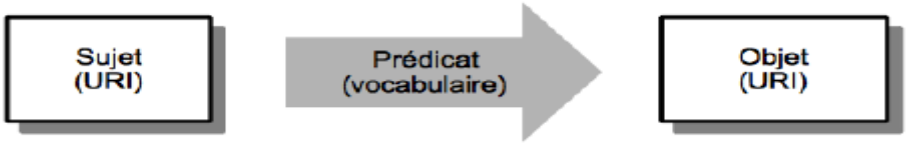


**Figure 02** : Schéma d'un triplet RDF [7]

***Le sujet*** d'un triplet est l'URI identifiant la ressource décrite.

***L'objet*** peut être soit une valeur littérale (une chaîne de caractères, un nombre ou une date), soit une URI d'une autre ressource qui est liée, d'une manière ou d'une autre, à l'objet.

***Le prédicat***, au milieu, indique le type de relation qui relie le sujet à l'objet (par exemple, le nom ou le numéro de sécurité sociale, pour une valeur littérale ; une connaissance ou un membre de la famille, pour une autre ressource). Il est, lui aussi, identifié par une URI.

Ces URI de prédicats proviennent des « vocabulaires ». Ce sont des collections d'URI qui peuvent être utilisées pour représenter l'information dans un certain domaine (par exemple FOAF, DC, etc.).

On distingue deux types principaux de triplets RDF : les triplets littéraux et les liens RDF.

- ***Les triplets littéraux*** ont un objet RDF littéral et servent à décrire les propriétés (ou attributs) d'une ressource.
- ***Les liens RDF*** permettent de modéliser des relations entre 2 ressources, et comprennent 3 références d'URI (2 pour le sujet et l'objet qui identifient les ressources liées, et une URI pour le prédicat qui définit le type de relation entre les ressources).

- Les liens RDF internes relient des ressources dans une seule source de données liées (les URI des sujets et des objets sont dans le même espace de noms) ;
- Les liens externes connectent des ressources dans des sources de données liées différentes (URI des sujets et objets des liens externes sont dans des espaces de noms différents).

Par exemple, si je veux exprimer l'assertion suivante «***Tim Berners-Lee est une personne***», elle correspond à la relation par le concept «***est***» des entités «***Tim Berners-Lee***» et du concept de «***personne***». Graphiquement, il peut être représenté ainsi :


**Figure 03** : Un exemple d'un modèle RDF

Si on remplace, chacun des signifiés par son signifiant sous la forme d'une URI :

| Sujet | Prédicat | Objet |
|---|---|---|
| <http://www.w3.org/People/Berners-Lee/card#i> | <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> | <http://xmlns.com/foaf/0.1/Person> |

**Figure 04** : Format d'une URI

Et sa représentation en syntaxe XML :

```
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#type"
xmlns:rdf="http://purl.org/dc/elements/1.1/">
<rdf:Description rdf:about="http:// dbpedia.org/resource/Bertrand_Delanoë ">
<dc:title> Bertrand_Delanoë </dc:title>
</rdf:Description>
</rdf:RDF>
```

Si, à présent, On souhaite exprimer l'assertion suivante «L'article Semantic Web a pour créateur Tim Berners-Lee», elle sera exprimée de la manière suivante :

| Sujet | Prédicat | Objet |
|---|---|---|
| <http://www.sciam.com/article.cfm?id=the-semantic-web> | <http://purl.org/dc/terms/creator> | <http://www.w3.org/People/Berners-Lee/card#i> |

Le modèle RDF propose donc une logique formelle (une phrase simple) pour encoder une donnée (étant entendue comme la relation entre deux signes) en s'appuyant sur les principes des triplets et de l'architecture du Web. Ainsi, chaque membre du triplet est une ressource qui peut-elle-même être le sujet ou l'objet d'autres assertions (le cas du prédicat est particulier de ce point de vue). Néanmoins, comme certains types de données ne sont pas forcément des entités, l'objet peut aussi être une chaîne de caractères, une date, un entier... ce qui est désigné par le terme «littéral». Par exemple, si je souhaite exprimer l'assertion «Tim Berners-Lee a pour nom "Tim Berners-Lee"», elle sera exprimée de la manière suivante :

| Sujet | Prédicat | Objet |
|---|---|---|
| <http://www.w3.org/People/Berners-Lee/card#i> | <http://xmlns.com/foaf/0.1/name> | "Tim Berners-Lee" |

**Le graphe :** Comme nous l'avons vu précédemment, un triplet peut être représenté sous la forme d'un graphe dont le sujet et l'objet sont les sommets et le prédicat un arc orienté. Ainsi, un triplet RDF est un graphe orienté.

La somme des triplets sur les différentes entités forme un graphe d'où l'expression «Giant Global Graph» utilisée pour la première fois en novembre 2007 par Tim Berners-Lee pour désigner

la somme de l'ensemble des triplets disponibles sur le Web. Par exemple, la représentation des triplets précédents forme le graphe suivant :

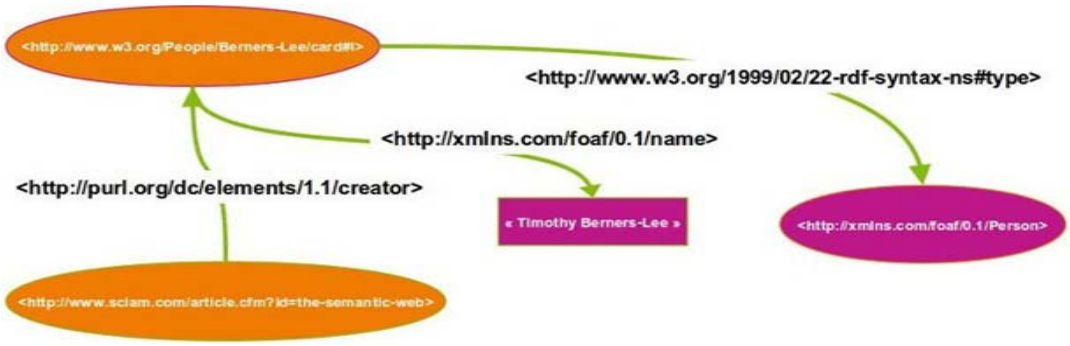


**Figure 05** : Graphe de plusieurs triplets RDF

## *I.4.3 RDF Schema :*

*RDF Schema* est un langage pour décrire des vocabulaires, des propriétés et des classes de ressources dans le modèle RDF. RDF Schema est une extension sémantique de RDF. Il fournit des mécanismes pour décrire des groupes de ressources similaires (classes) et des relations entre ces ressources (propriétés). Les descriptions de vocabulaire de RDF Schema sont écrites en RDF en utilisant les termes (primitives) décrits dans la spécification du schéma RDF[1]. La combinaison de RDF Schema et RDF est souvent référencée par RDF(S). Autrement dit, RDF(S) fournit un moyen pour décrire les types des ressources et leurs caractéristiques.

```
<rdf:RDF
xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:xsd ="http://www.w3.org/2001/XMLSchema#"
xml:base ="http:// dbpedia.org/resource/nom #">
<rdfs:Class rdf:about="Article">
<rdfs:label xml:lang="en-US">l'article scientifique</rdfs:label>
</rdf:Class>
<rdfs:Property rdf:about="title">
<rdfs:label xml:lang="en-US">titre de l'article scientifique</rdfs:label>
<rdfs:domain rdf:resource="Article"/>
```

[1] http://www.w3.org/TR/rdf-schema/

<rdfs:range rdf:resource="&xsd;string"/>
</rdf:Class>
<rdfs:Property rdf:about="pageNum">
<rdfs:label xml:lang="en-US">nombre de pages</rdfs:label>
<rdfs:domain rdf:resource="Article"/>
<rdfs:range rdf:resource="&xsd;integer"/>
</rdf:Class>
</rdf:RDF>

### *I.4.4 SPARQL :*

SPARQL (Simple Protocol and RDF Query Language) : Langage d'interrogation des ontologies représentées sous forme de graphes RDF/S. Il est pour les bases de connaissances RDF ce que SQL est pour les bases de données relationnelles [8].

SPARQL est l'équivalent de Structured Query Language (SQL) pour les bases de données relationnelles, dont, d'ailleurs, la syntaxe y ressemble grandement. Des opérations de sélection (SELECT), d'ajout (INSERT), de mise à jour (UPDATE) et d'effacement (DELETE) peuvent être utilisées. Une requête est composée de plusieurs parties. Dans un premier temps, on spécifie les vocabulaires utilisés. Ensuite, on indique l'opération à effectuer (sélection, ajout, etc.).
Pour finir, il est possible d'utiliser des opérateurs ensemblistes (union, projection, ...) pour réduire ou grouper des ressources.

SPARQL désigne à la fois le langage de requête pour RDF et le service Web qui permet de soumettre une requête. Ce langage très simple fonctionne essentiellement par filtrage de motifs sur des graphes et s'inspire de la syntaxe de SQL et de N3. Par exemple, voici une requête valide sur DBpedia permettant de retrouver les Grandes Écoles parisiennes et leur nombre d'élèves (ces informations sont disponibles dans Wikipedia donc dans DBpedia) :

Exemple :

**SELECT DISTINCT** ?ecole ?nombreeleves
**WHERE** {?ecole <http://www.w3.org/2004/02/skos/core#subject>
<http://dbpedia.org/resource/Category:Grandes_Écoles>.
?ecole <http://dbpedia.org/ontology/city>
<http:// dbpedia.org/resource/Paris>.
?ecole <http://dbpedia.org/ontology/numberOf
Students> ?nombreeleves}

## I.5 EVOLUTION DU WEB DE DONNEES :

Les principes du Web de données se retrouvent appliqués dans Linking Data Project[2].
Le projet, fondé en 2007 et supporté par le W3C, a comme objectif d'exploiter les informations contenues dans le Web, en les récupérant des basés de données existantes, en les convertissant au format RDF et en les rendant disponibles sur le Web. L'objectif originel de ce projet, qui a engendré une communauté active et toujours en expansion, était de commencer le Web des données par une identification des jeux de données existants et accessibles sous des licences ouvertes, de les convertir en RDF conformément aux principes des données liées et de les publier sur le Web. Le projet a toujours été ouvert à quiconque publie conformément aux principes des données liées. Cette ouverture est un facteur-clé dans le succès du démarrage du projet.

Les Figures 5,6, 7 et 8 montrent l'expansion du nombre de données publiées sur le Web en tant que données liées depuis les origines du projet Linking Open Data. Chaque nœud dans le diagramme représente un jeu distinct de données publiées en tant que données liées. Les arcs indiquent l'existence de liens entre deux jeux.

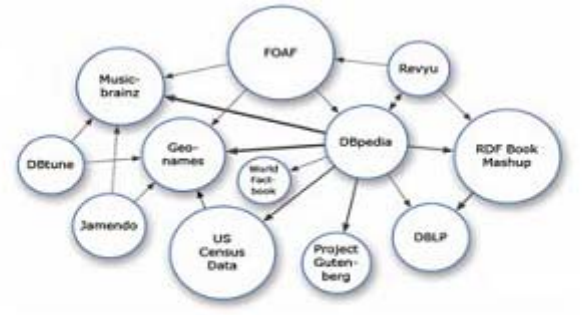


**Figure 06** : Linked data en mai 2007

[2] http ://linkeddata.org/

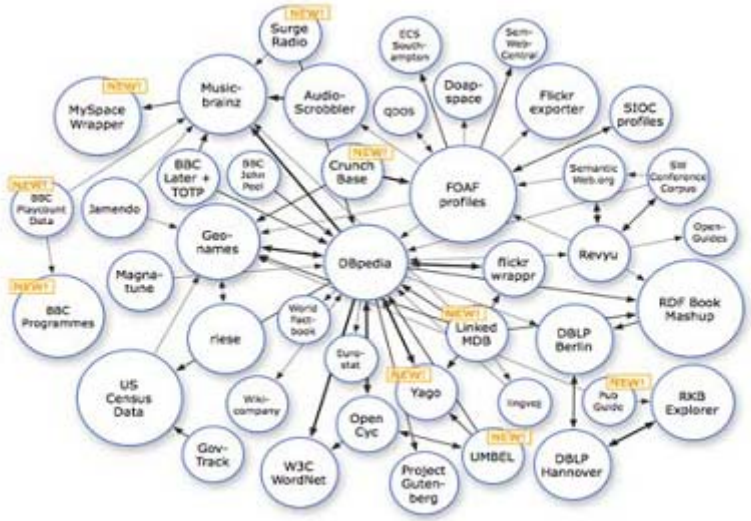


**Figure 07** : Linked data en septembre 2008

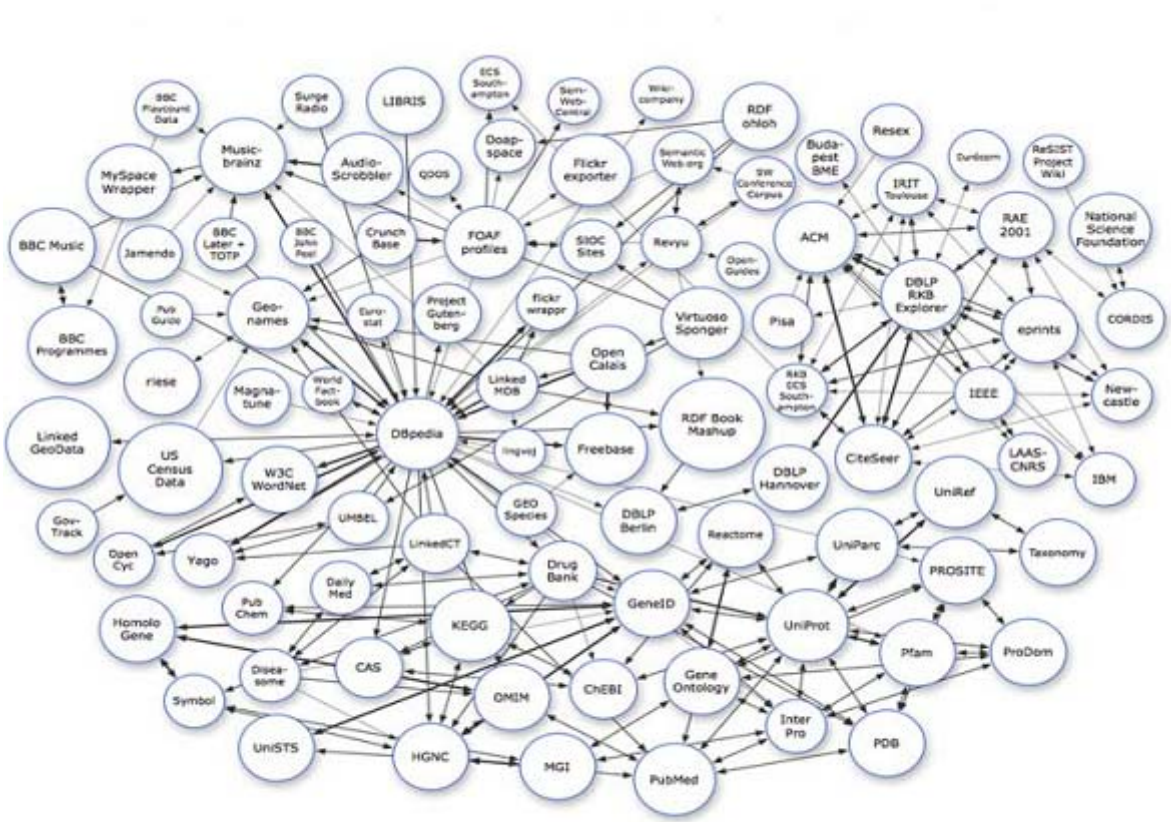


**Figure 08** : Linked data en Juillet 2009

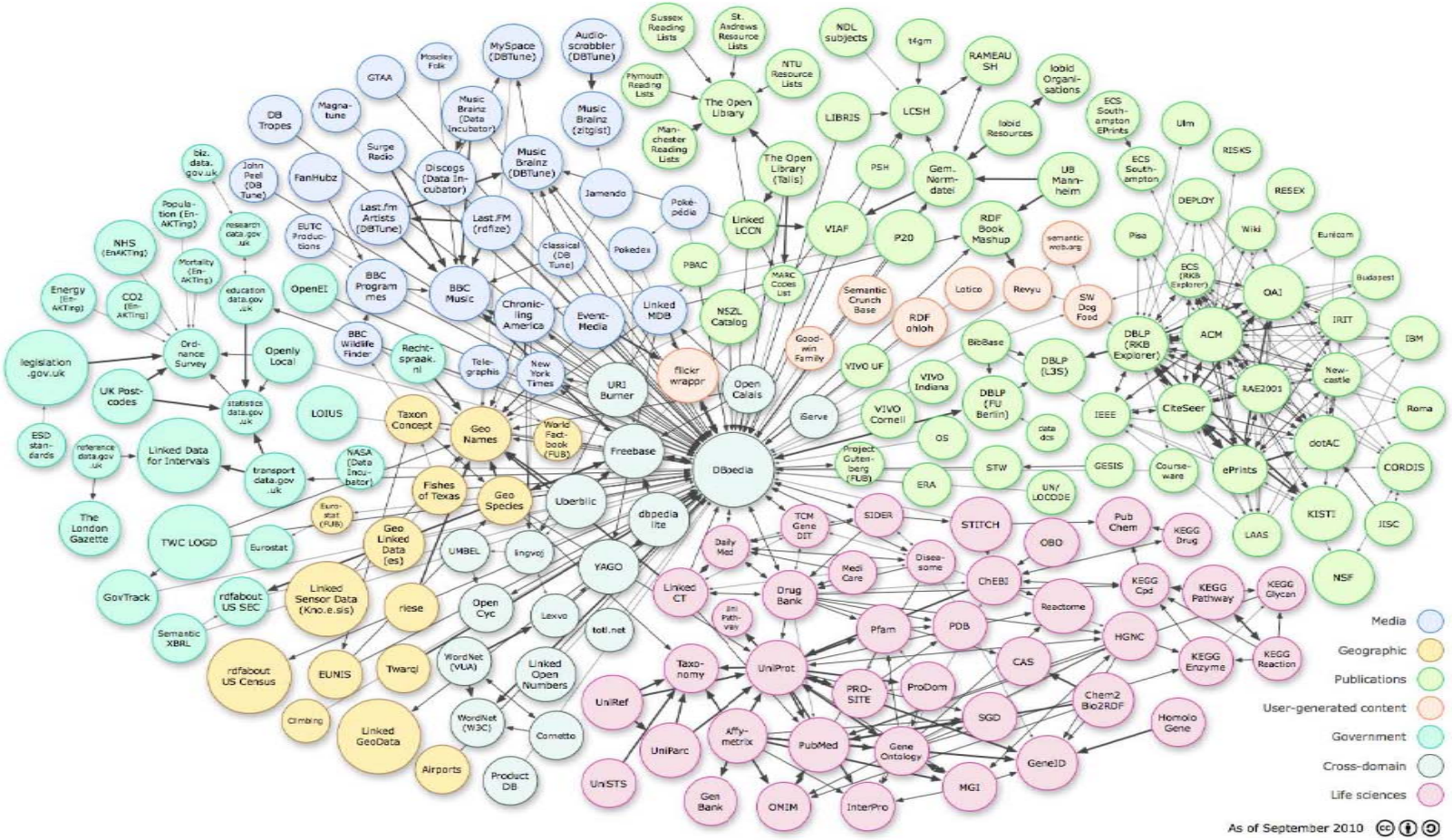


**Figure 09** :Evolution du web de données 2010[3]

[3] http://lod-cloud.net/

Depuis quelques années donc, les efforts de cette communauté ne cessent pas de s'intensifier, ce qui a porté une croissance remarquable des contenus disponibles. La rapide évolution est principalement due à la nature open source du projet, permettant à toute communauté d'alimenter et agrandir les données.

La situation en septembre 2011 comptait plus que 31 milliards de triplets RDF, c'est-à-dire 295 bases mises en relation, traitant de domaines différents. Les "connexions" sont exprimées dans le graphe par les arcs entre datasets : elles peuvent être bidirectionnelles ou unidirectionnelles selon que les liens existent dans les deux bases de données ou pas.

## I.6 OUTILS DE PUBLICATIONS :

Plusieurs outils ont été développés jusqu'à présent pour l'exploitation des Linked Data. Nous en citons ici quelques uns [10] :

_ Plateformes de publications de données. Ces outils permettent de publier les données au format RDF et de les interroger avec des services Web SPARQL. Parmi les plus connus, on retrouve :

– D2R Server: http://www4.wiwiss.fu-berlin.de/bizer/d2r-server/

– Linked Data API http: //purl.org/linked-data/api/spec

– Linked Media Framework http: //code.google.com/p/kiwi/

– Joseki http://www.joseki.org/

_ Editeurs et Validateurs des données RDF, comme :

– Hyena http://hypergraphs.de/

– Vapour http://validator.linkeddata.org/

_ Moteurs de recherches sémantiques pour la recherche des données RDF, tels que :

– Swoogle : http://swoogle.umbc.edu/

– Watson http://watson.kmi.open.ac.uk/WatsonWUI/

– SameAs: http://sameas.org/

## I.7 CONCLUSION:

On a présenté les concepts et les principes de base des données liées, un aperçu des technologies qui les supportent telles qu'URI, HTTP et RDF. Ensemble, ces technologies et ces principes permettent d'élaborer un style qui tisse des données dans la toile même d'Internet, une caractéristique unique qui expose les données liées au potentiel rigoureux et sans limites du Web au sens large.

Il est également intéressant de noter que de grandes entreprises du Web sont déjà en train de construire ces espaces de données. Google, Yahoo! et Facebook ont commencé à relier les utilisateurs, les données géographiques et commerciales, et à utiliser ces espaces dans leurs applications.

# *CHAPITRE II :*

# *LES BASES DE DONNEES RELATIONNELLES*

## II.1 INTRODUCTION :

Le modèle relationnel a été introduit pour la première fois par Ted Codd du centre de recherche d'IBM en 1970 dans un papier désormais classique XXX, et attira immédiatement un intérêt considérable en raison de sa simplicité et de ses fondations mathématiques. Le modèle utilise le concept de relation mathématique qui est associée à une table de valeurs comme primitive, et ses bases théoriques reposent sur la théorie des ensembles et sur la logique du premier ordre.

Au début des années 70, le modèle relationnel fait son apparition .La recherche se passionne : impossible de nier les progrès apportés concernant la représentation et la manipulation des données par les systèmes. Dix ans passent, les spécialistes déchantent : ce top-model engendre en définitive des systèmes commerciaux bien moins performants que leurs concurrents fondés sur les modèles réseau ou hiérarchique.

Deux ans plus tard et voilà que les produits relationnels peuvent prétendre relayer les "vieux" systèmes. Leurs apports sont fondamentaux : les nouvelles fonctionnalités permettent un confort d'utilisation sans précédent. Les systèmes commerciaux s'emparent des concepts de ce nouveau modèle. Celui-ci, désormais, s'impose [11].

## II.2 DEFINITION D'UNE BASE DE DONNEES :

Une base de données est un gros ensemble d'informations structurées mémorisées sur un support permanent qui peut être partagée par plusieurs applications et qui est interrogeable par le contenu [12].

Il existe 4 types de bases de données :

1. BD Hiérarchiques : les plus anciennes fondées sur une modélisation arborescente des données.
2. BD Relationnelles : organisation des données sous forme de tables et exploitation à l'aide d'un langage déclaratif (ex : Oracle, mySQL, Access).
3. BD Déductives : organisation de données sous forme de table et exploitation à l'aide d'un langage logique.
4. BD Objets : organisation des données sous forme d'instances de classes hiérarchisées qui possèdent leurs propres méthodes d'exploitation.

## II.3 CONCEPTION D'UNE BASE DE DONNEES RELATIONNELLE :

La modélisation se réalise en trois étapes principales qui correspondent à trois niveaux d'abstraction différents :

**Niveau conceptuel** : représente le contenu de la base en termes conceptuels, indépendamment de toute considération informatique.

**Niveau logique relationnelle** : résulte de la traduction du schéma conceptuel en un schéma propre à un type de BD.

**Niveau physique** : est utilisé pour décrire les méthodes d'organisation et d'accès aux données de la base.

### II.3.1 Modélisation conceptuelle

Le niveau central est le niveau conceptuel. Il correspond à la structure canonique des données qui existent dans l'entreprise, c'est-à-dire leur structure sémantique inhérente sans souci d'implantation en machine, représentant la vue intégrée de tous les utilisateurs. La définition du schéma conceptuel d'une application n'est pas un travail évident. Ceci nécessite un accord sur les concepts de base que modélisent les données [11].

Les éléments de base du modèle ER (Entité-Relation)ou E-A (Entité -Association)

1. **Entité** : définit comme un objet pouvant être identifie distinctement.
   Il existe deux catégories d'entités :
   - Entités régulières : son existence ne dépend pas de l'existence d'une autre entité.
   - Entités faibles : son existence dépend de l'existence d'une autre entité.
2. **Attributs** : caractéristiques ou propriétés des entités. Un attribut peut être obligatoire ou facultatif et avoir un domaine de valeurs.
3. **Les relations** : représentent les liens existants entre les entités. Contrairement aux entités, les relations n'ont pas de relations propres. Les relations sont caractérisées, comme les entités, par un nom et éventuellement des attributs.
4. **Cardinalité** : la description complète d'une relation nécessite la définition précise de la participation des entités. La cardinalité est le *nombre de participation d'une entité à une relation*.

Les cardinalités sont appelées **cardinalités maximales** dans la mesure où elles représentent le nombre maximum de participations d'une entité à une relation.

En revanche, la **cardinalité minimale** est le nombre minimal de participations d'une entité à une relation. La cardinalité minimale peut être 0 ou 1.

Les cardinalités maximales et minimales traduisent les contraintes propres aux entités et relations. Dans un schéma conceptuel, elles sont représentées comme suit :

**0-1** aucune ou une seule

**1-1** une et une seule

**0-N** aucune ou plusieurs

**1-N** une ou plusieurs

5. **L'identifiant** : parmi tous les attributs de l'entité, l'identifiant est un attribut ou un ensemble d'attributs permettant de déterminer une et une seule entité à l'intérieur de l'ensemble. Graphiquement les identifiants sont les attributs soulignés. L'entité faible aura un identifiant composé de l'identifiant de l'entité dont elle dépend et d'un autre attribut.

## II.3.2 Modélisation logique relationnelle :

Dans le modèle relationnel, les entités du schéma conceptuel sont transformées en tableaux à deux dimensions. Le modèle relationnel s'appui sur trois concepts fondamentaux : le domaine, l'attribut et la relation ou table.

1. **Domaine** : ensemble de valeurs défini en extension ou en intension. Un domaine peut être simple ou composé.

Domaine simple : si tous les éléments sont atomiques ou décomposables.

Ex : l'ensemble des grades du salarié peut être définit en extension par employé, agent de maîtrise, ou cadre.

Domaine composé : si les éléments peuvent être décomposés.

Ex : les dates sont décomposées d'un jour, un mois et une année.

2. **Attribut** : chaque colonne est appelée attribut et contient un ensemble des valeurs d'un domaine. Chaque ligne représente un *tuple*.
3. **Relation ou table** : une relation est un tableau à deux dimensions. Le degré de la relation est le nombre de colonnes ou des domaines considérés.

### II.4 LES CONTRAINTES D'INTEGRITES :

Permettent d'assurer la cohérence des données. Les contraintes d'intégrité sont :

Contrainte de domaine : restriction de l'ensemble des valeurs possibles d'un attribut.

Contrainte de clé : définit un sous-ensemble minimal des colonnes tel que la table ne puisse contenir deux lignes ayant mêmes valeurs pour ces colonnes.

Il existe trois types de clés :

- Clé primaire : Ensemble minimum d'attributs qui permet de distinguer chaque n-uplet de la table par rapport à tous les autres. Chaque table doit avoir une clé primaire.
- Clé candidate : Ensemble minimum d'attributs susceptibles de jouer le rôle de la clé primaire.
- Clé étrangère : fait référence à la clé primaire d'une autre table.

## II.5 LA THEORIE DE LA NORMALISATION :

Permet de définir formellement la qualité des tables au regard du problème posé par la redondance des données. La théorie de la normalisation s'appui sur la dépendance fonctionnelle. Cod a définit un ensemble de formes normales caractérisant les tables relationnelles :

· ***Première forme normale*** : si elle ne contient que des attributs atomiques.

· ***Deuxième forme normale*** : si elle ne contient que des attributs atomiques et, si de plus, il n'existe pas de dépendance fonctionnelle entre une partie d'une clé et une colonne non clé de la table.

· ***Troisième forme normale*** : si elle ne contient que des attributs atomiques, s'il n'existe pas de dépendance fonctionnelle entre une partie d'une clé et une colonne non clé de la table et si, de plus, aucune dépendance fonctionnelle entre les colonnes non clé.

Ainsi, plus une table est normalisée moins elle comporte de redondances et donc de risques d'incohérence sémantiques dans les schémas relationnels.

**Règles à suivre pour concevoir un schéma relationnel**

Les règles principales de transformation d'un schéma conceptuel Entité-Relation en un schéma relationnel sont :

**Règle I** : Toute entité est traduite en une table relationnelle dont les caractéristiques sont les suivantes :

- le nom de la table est le nom de l'entité ;
- la clé de la table est l'identifiant de l'entité ;
- les autres attributs de la table forment les autres colonnes de la table.

**Règle II** : Toute relation binaire plusieurs à plusieurs est traduite en une table relationnelle dont les caractéristiques sont les suivantes :

- le nom de la table est le nom de la relation ;

- la clé de la table est formée par la concaténation des identifiants des entités participant à la relation ;
- les attributs spécifiques de la relation forment les autres colonnes de la table.

Une contrainte d'intégrité référentielle est générée entre chaque colonne clé de la nouvelle table et la table d'origine de cette clé.

**Règle III** : Toute relation binaire un à plusieurs est traduite :

- soit par un report de clé : l'identifiant de l'entité participant à la relation côté N est ajoutée comme colonne supplémentaire à la table représentant l'autre entité. Cette colonne est parfois appelée *clé étrangère*. Le cas échéant, les attributs spécifiques à la relation sont eux aussi ajoutés à la même table ;
- soit par une table spécifique dont les caractéristiques sont les suivantes :

- le nom de la table est le nom de la relation ;
- la clé de la table est l'identifiant de l'entité participent à la relation côté 1 ;
- les attributs spécifiques de la relation forment les autres colonnes de la table.

**Règle IV** : Toute relation binaire un à un est traduite, au choix, par l'une des trois solutions suivantes :

- fusion des tables des entités qu'elle relie (choix1) ;
- report de clé d'une table dans l'autre (choix2) ;
- création d'une table spécifique reliant les clés des deux entités (choix3).

Les attributs spécifiques de cette relation sont ajoutés à la table résultant de la fusion (choix1), reportés avec la clé (choix2), ou insérés dans la table spécifique (choix3).

## II.6 LE LANGAGE SQL

Le langage SQL (Structured Query Langage) s'appui sur les opérateurs de l'**algèbre relationnelle** définit en 1970 par Codd, mathématicien, chercheur chez IBM. Le langage SQL est basé sur le concept de relation de la théorie des ensembles.

1-Opérateurs de l'algèbre relationnelles :

a- Les opérations de base :

- La projection
- La sélection
- La Jointure

b- Les opérations ensemblistes

- L'union

- L'intersection
- La différence
- Produit cartésien

## II.7 CONCLUSION :

Depuis l'avènement des bases de données qui date des années, ces dernières ont bien marqué leur présence dans tous les domaines à savoir les systèmes d'information, les technologies mobiles et le web sémantique.

Les bases de données constituent actuellement le pilier de toutes applications notamment celles qu'on trouve sur le Net.

# *Deuxième partie*

# *Etat de l'art*

# *CHAPITRE III :*

# *METHODES ET OUTILS DE TRANSFORMATION DES BDDS RELATIONNELLES EN LINKED DATA TRAVAUX CONNEXES*

## III.1 INTRODUCTION :

Dans ce chapitre, nous allons présenter l'état actuel de l'art des méthodes et des paradigmes concernant la transformation de données relationnelles en données liées. Nous allons d'abord présenter les travaux qui existent par ordre chronologique, à la fin certaines réalisations comme serveur D2R et Openlink Virtuoso serveur parmi d'autres. Pour conclure ce chapitre, nous allons proposer une étude comparative des différentes approches déjà cités de l'état de l'art.

## III.2 OUTILS DE TRANSFORMATION DES BDD RELATIONNELLES EN LINKED DATA :

### III.2.1 OPENLINK VIRTUOSO UNIVERSAL SERVER: Open source 1999

#### III.2.1.1 Définition:

Openlink Virtuoso Universal Server acte comme un moteur virtuel d'appariement de nombreux types de base de données comme DB2, SQL Server, Oracle, Sybase, etc. En plus, pour fournir une interface unique pour les différentes bases de données, Openlink Virtuoso Universal Server supporte les Web et les standards d'accès aux données : les technologies XML (XPath expressions, XSLT, XML storage), les technologies Web services (WSDL, UDDI, SOAP), WebDAV, SMTP, JDBC et ODBC etc.

Virtuoso donne une vue données liées de données relationnelles, cette procédure est faite à l'aide de « Virtuoso's declarative Meta Schema language » construit au-dessus de SPARQL, pour l'appariement de données SQL vers des Ontologies RDF, par la suite ces dernier serrant interrogés avec SPARQL [15].

#### III.2.1.2 Objectif :

Il est complètement transparent pour un utilisateur qui peut facilement accéder aux données distribués sur différentes serveurs hétérogènes [3].

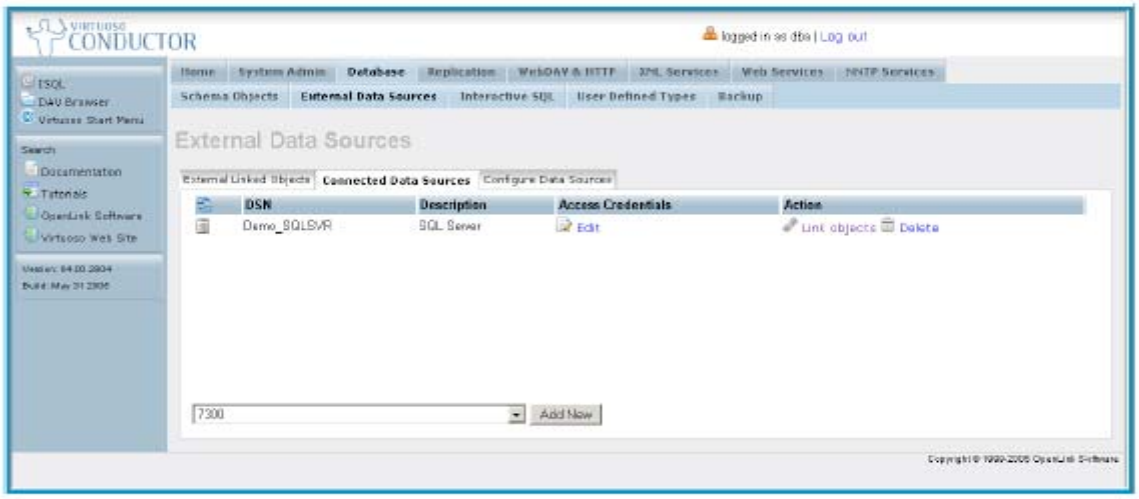


**Figure 10** : Interface de VIRTUOSO [16]

### III.2.2 D2R SERVER: ChristianBizer and RichardCyganiak 2006

#### III.2.2.1 Définition :

Le D2R Server est un projet académique open source développé à l'université Freie de Berlin. Il a été initié fin 2006 (traduit en premier Novembre 2006).

Il fournit un environnement intégré avec de multiples options pour accéder aux données relationnelles en utilisant différentes méthodes telles que le point de terminaison SPARQL, les données liées (négociation de contenu HTTP 303 déréférencement), RDF décharge, et l'accès de l'API à base de Jena (appels d'API sont réécrits pour SQL).

D2RQ prend en charge les mapping directs et domaine sémantique. Le langage déclaratif de DR2Q mapping est formellement défini par un schéma RDFS. Il est le successeur de la D2R CARTE (langage basé sur XML). Les mappages sont exprimés en RDF, mais aussi en grande partie s'appuient sur des fragments SQL pour exprimer certains conditions ou d'utiliser des fonctions d'agrégation. Ontologies existantes (RDFS / OWL) peuvent être réutilisés en vue d'intégrer la sémantique de domaine dans le processus de mapping. Le mapping automatique génère un mapping direct des D2RQ qui reflète le schéma de base de données, créant ainsi une ontologie locale. Cette correspondance directe peut être personnalisée manuellement. Eventuellement, le mapping direct généré peut suivre les règles proposées dans la spécification de mapping direct du W3C.

### III.2.2.2 Objectif :

D2R Server est un outil qui permet de servir une base de données relationnelle sous forme de données liées.

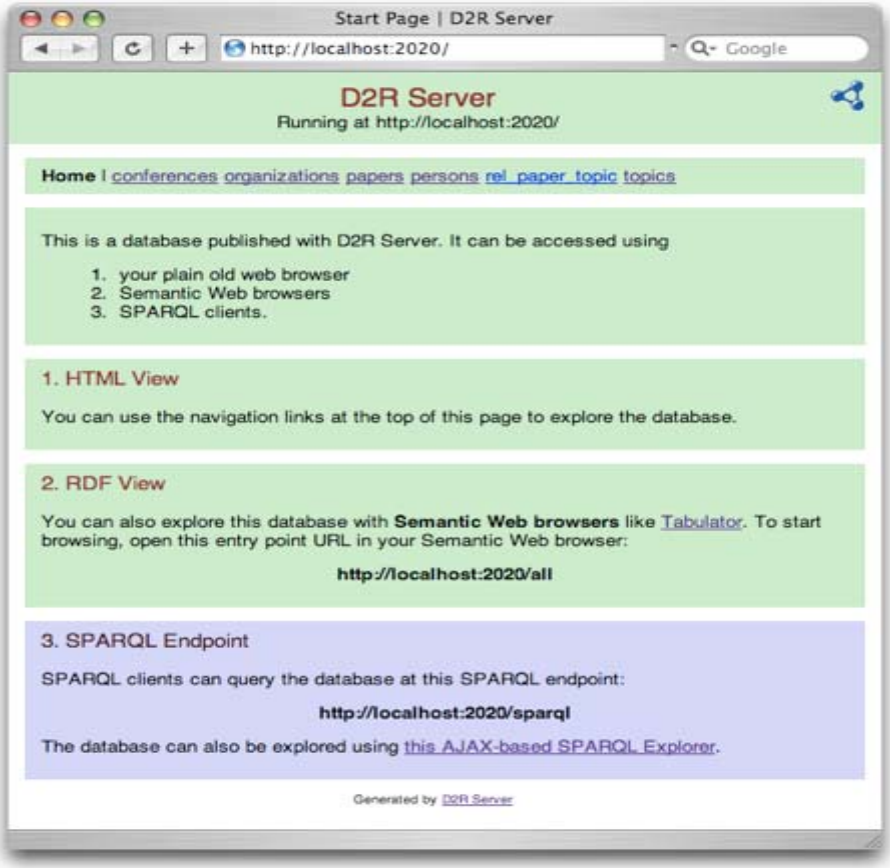


**Figure 11** : interface de D2RServer

### III.2.3 TRIPLIFY: Sören Auer, Sebastian Dietzold, Jens Lehmann,Sebastian Hellmann, David Aumueller, 2009.

### III.2.3.1 Définition :

Étant donné que de nombreux cas de ces applications Web sont déployés sur Internet, en les aidants à exposer rapidement leur base de données pour aboutir à un coup de pouce de l'adoption du Web sémantique. Triplify est une approche simple mais agréable, pour être utilisé comme un poids léger, facile à apprendre, plug-in pour les applications Web existantes. Il est basé sur la mapping des requêtes HTTP URI sur les requêtes à la base de données relationnelle. Par rapport à la méthode de mapping direct, Triplify prend le problème dans l'autre sens : il se concentre d'abord sur les données relationnelles qui importent réellement, au lieu de traduire le schéma relationnel dans une ontologie ad hoc, puis remplit cette ontologie. Les mapping sont mises en œuvre dans les instructions SQL imbriquées dans les scripts PHP, par conséquent, toute construction SQL ou une

fonction d'agrégation peuvent être utilisés. Auteurs affirment que "Triplify facilite la création de moteurs de recherche sur mesure ciblée à certains créneaux, par exemple la recherche de contenu spécifique dans divers blogs, wikis, forums ou". Il faut également mentionner que certains modules Triplify traitent spécifiquement de la génération de métadonnées de provenance (en utilisant le Provenance Vocabulary) ainsi que la publication de la mise à jour des journaux.

**III.2.3.2 Objectif :**

Le but de Triplify est de permettre aux applications Web populaires (comme les applications de blog ou CMS) de publier le contenu de leur base de données relationnelle en RDF ou de données liées.

### III.2.4 DIRECT MAPPING : W3C 2009

**III.2.4.1 Définition :**

Un appariement directe prend en entrée un schéma relationnelle et une instance de ce schéma (ou simplement une base de données relationnelle) et il retourne en sortie un graphe RDF. Nous pouvons requêter sur le résultat avec le langage SPARQL ou n'importe quel langage d'interrogation de graphes RDF. A ce moment Direct mapping est sortie de la phase d'étude est-elle est candidat d'être une recommandation de W3C, ce qui veut dire que les développeurs conceptuels de cette approche pense que la version courante est tellement stable qu'ils peuvent encourager les développeurs à implémenter le standard.

La phase de transformation de cette approche est simple, pour tous tuples de tous tables dans la base, la même routine sera exécutée, le résultat finale c'est d'obtenir une ensemble de triplets (S.P.O).

Le sujet (S) est obtenu par l'utilisation de la clé de tuple, il est agrégé par un URI de base. Les auteurs de cette approche utilisent le nom de table, nom de colonne clé pour former l'URI.

Les prédicats (P) sont obtenus par un mécanisme similaire, le nom de colonne et concaténé avec le nom de table et l'URI de base.

Les objets (O) sont considérés directement comme littéraux.

### III.2.5 R2RML :W3C 2012

**III.2.5.1 Définition :**

R2RML (Relational To RDF Mapping Language) est un langage définit par le Consortium W3C.Maintenant, R2RML est un candidat pour être une recommandation de l'équipe de recherche de W3C. La première version de ce langage a été réalisée en octobre 2010, cette version a été subi

a trois (03) mise à jours, la première en 2011 et la dernière en février 2012 et c'est la version courante.

Celant les constructeurs de cette approche, l'appariement d'une base de données relationnelle avec une autre liée (RDF) est écrit dans un fichier « texte » syntaxiquement et sémantiquement correcte en suivant la spécification « R2R mapping laguage », ce texte sera écrit en suivant l'annotation « turtle ». Le fichier d'appariement lui-même est un graphe RDF et on peut l'utiliser par la suite (par exemple lors de l'extraction des informations). En plus, le formalisme RDF n'est utilisé seulement comme modèle de données en sortie mais aussi pour représenter l'appariement (mapping).

**III.2.5.2 Objectif :**

R2RML est créé pour l'expression de l'appariement personnalisé depuis les bases de données relationnelle vers les données liées (RDF).

Ces règles d'appariement permettent d'exporter les données relationnelles vers un modèle RDF dans une structure et avec un vocabulaire approprié.

**III.2.6 DB2TRIPLES :**

**III.2.6.1 Définition :**

DB2Triples est une implémentation de la R2RML W3C et les recommandations direct mapping . Il a est développé par la société Antidot dans le cadre d'une suite de grand logiciels. DB2Triples est livré comme une bibliothèque (library) Java, disponible sous les termes de la licence LGPL licence open source, et validé avec MySQL et PostgreSQL back-ends. Il prend en entrée un document R2RML, une connexion de base de données et une requête SPARQLs(parql querry), et renvoie les résultats en RDF / XML, N3, N-Triples ou turtle. Par conséquent, il est en mesure de traiter les requêtes SPARQL, mais ce n'est pas un point d'extrémité SPARQL, en mesure de recevoir des demandes sur HTTP.

**III.2.7 UITRAWRAP** :

**III.2.7.1 Définition :**

Ultrawrap est un système de développement de RDB2RDF.En outre, les utilisateurs peuvent créer manuellement personnalisé mappages à l'aide du langage de mapping R2RML.L'architecture unique de Ultrawrap permet des requêtes SPARQL pour être optimisés par le moteur SQL.

Par conséquent, le temps d'exécution d'une requête SPARQL est comparable à son équivalent SQL sémantiquement interroge le temps d'exécution.

Ultrawrap se compose de deux éléments : Ultrawrap compilation et Ultrawrap Server.

Ultrawrap Compile s'exécute uniquement au début de l'exécution, puis Ultrawrap Server continue et met en place l'extrémité SPARQL sur une jetée Server [14].

### III.2.7.2 Objectif :

Ultrawrap mappe automatiquement directement base de données relationnelle schéma dans une ontologie OWL et expose le contenu de bases de données relationnelles comme RDF et par un Extrémité SPARQL utilisant le Direct Mapping W3C.

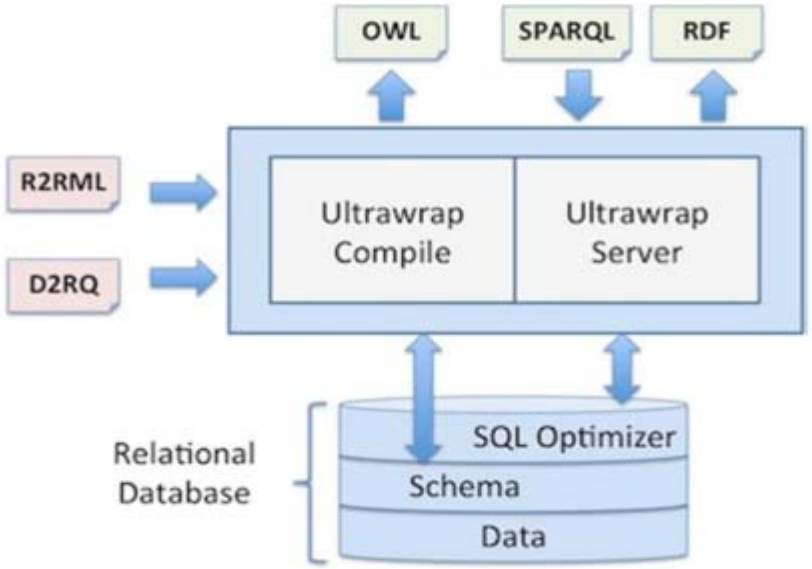


**Figure 12** : Schéma de Ultrawrap

## III.3 ETUDE COMPARATIVE:

Dans ce chapitre, nous avons examiné les (non-exhaustive) des normes et les implémentations qui définissent des méthodes pour transformer les données relationnelles en RDF. Chacun de ceux-ci ont leurs avantages et leurs inconvénients. Par conséquent, nous vous proposons un tableau comparatif en conservant les solutions existantes précitées de l'état de l'art, qui résume leurs principales caractéristiques.

### III.3.1 Tableau comparative :

| **Approches** | OpenLink Software, 1999 | Chris Bizer, Richard yganiak, Jörg arbers,2009 | Sören Auer, Sebastian Dietzold,Jens Lehmann,Sebastian Hellmann,David Aumueller,2009 | Marcelo Arenas, Alexandre Bertails, ric Prud'hommeaux, Juan Sequeda, 2012 **Direct mapping** | Richard Cyganiak, 2012 **R2RML** | Société Antidot 1999 | Juan Sequeda. Daniel P. Miranker, Rudy Depena, 2013 |
|---|---|---|---|---|---|---|---|
| **Source d'entrée** | BDD | BDD | BDD | BDD | BDD | Un document R2RML | R2RML+D2RQ |
| **Sortie** | RDF | Fichier mapping | RDF | Fichier mapping | Fichier mapping | RDF | RDF |
| **Langage de requête** | SPARQL | SPARQL | SPARQL | SPARQL | SPARQL | SPARQL | SPARQL |
| **Outils** | **Openlink Virtuoso Universal Server** | **D2R Server** | **Triplify** | | | **DB2Triples** | **Ultrawrap** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Licence** | Open source | Apache v.2.0 | Open source | | | Open source | Commercial |
| **Exécution** | logiciel | Via le web | Via le web | | | logiciel | Via le web |
| **Site web** | http://downloads.sourceforge.net/virtuoso/virtuoso-opensource-6.1.8.tar.gz | http://d2rq.org/d2r-server | http://triplify.org/triplify | http://www.w3.org/TR/rdb-direct-mapping/ | http://www.w3.org/ns/r2rml | http://github.com/antidot/db2triples | |
| **Mise a jour**<br>**Versionnement** | oui | non | Non | Oui | Oui | Oui | Non |

**Tableau 01** : Tableau comparative

## III.3.2 Résumé du tableau comparatif :

Dans ce tableau, nous avons récapitulé toutes les outils ainsi que les approches permettant un accès au web de données. Ce tableau a été récapitulé selon plusieurs critères parmi eux les fondateurs des approches ou des outils, les sources d'entrées et les résultats sortante, les outils qu'ils travaillent avec, les mises a jours, etc…

## III.4 Conclusion :

Dans ce chapitre, nous avons développé l'état de l'art concernant la transformation de données relationnelles en RDF. Nous avons abordé différentes implémentations de logiciels tels que OpenLink Virtuso Universal Server, D2R Server, db2triples et Triplify et des normes théoriques développés par le W3C ainsi que le direct mapping et R2RML. Nous avons conclu cet Etat de l'art en comparant ces différentes approches.

# *CHAPITRE IV :*

# *IMPLEMENTATION*

## IV.1 INTRODUCTION :

Dans ce chapitre, nous allons aborder la mise en œuvre effective de l'approche Tuple2Triple

Pour la réalisation de ce projet nous avons utilisé pour l'essentiel des logiciels libres. La création de la base de données est effectuée avec MYSQL et nous avons utilisé un logiciel de Java Netbeans 7.4, ainsi que les règles de passages appliqué pour aboutir à un fichier RDF et RDFs.

Enfin, nous présenterons les résultats de la traduction obtenue avec la mise en œuvre et une analyse de la performance.

## IV.2 OUTIL ET LOGICIEL UTILISE :

### IV.2.1 WampServer (Version2.2) :

***WampServer*** (anciennement **WAMP5**) est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL.

### IV.2.2 MYSQL :

Pour la création de la base de données, nous avons utilisé MySQL qui est un système de gestion de bases de données relationnelles. Le SQL dans "MySQL" signifie "Structured Query Language" : le langage standard pour les traitements de bases de données. MySQL est Open Source. Le logiciel MySQL est un serveur de base de données SQL.

Le logiciel MySQL est défini http://www.mysql.com/

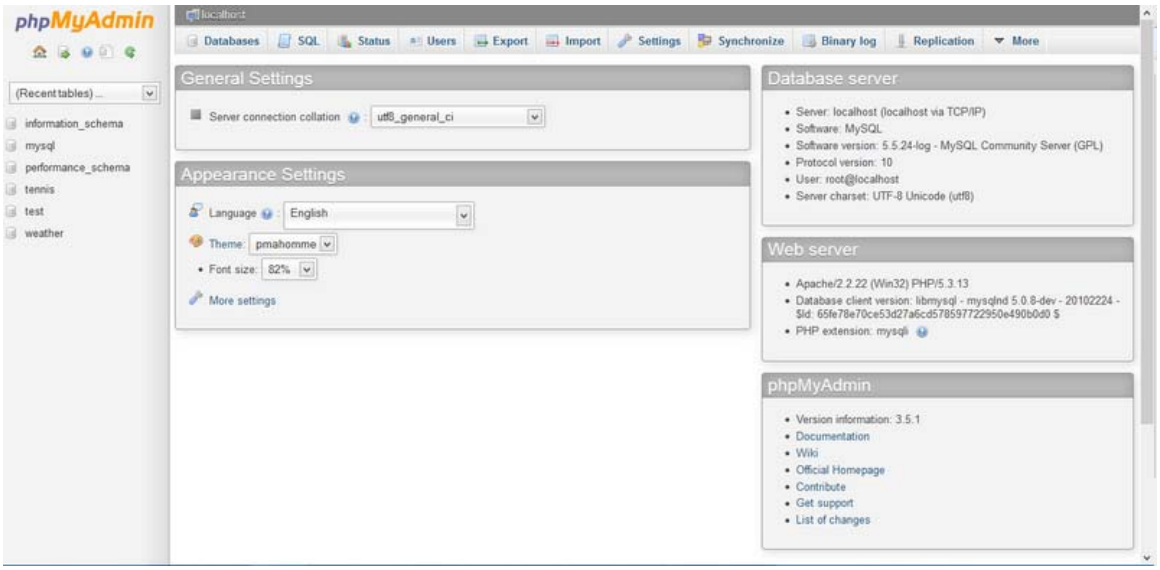


**Figure 13 :** Administration Web des bases MySQL

### IV.2.3 Netbeans :

Pour la création du logiciel, nous avons utilisé Netbeans 7.4 qui est un environnement de développement intégré (IDE) pour Java, placé en open source par Sun en juin 2000 sous licence CDDL (Common Développent and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages, comme Python, C, C++, XML et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages web).

NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), Mac OS X et Open VMS.

Un environnement Java Développent Kit JDK est requis pour les développements en Java

NetBeans est lui-même développé en Java, ce qui peut le rendre assez lent et gourmand en ressources mémoires [20].

Le logiciel NetBeans est défini www.netbeans.org


**Figure 14 :** NETBEANS Version7.4

### IV.2.4 Java

Qui est à la fois un langage de programmation informatique orienté objet et un environnement d'exécution informatique portable créé par James Gosling et Patrick Naughton employés de Sun Microsystems avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.


**Figure 15 : J**ava

Java est à la fois un langage de programmation et un environnement d'exécution. Le langage Java a la particularité principale que les logiciels écrits avec ce dernier sont très facilement portables sur plusieurs systèmes d'exploitation tels que Unix, Microsoft Windows, Mac OS ou Linux avec peu ou pas de modifications... C'est la plate-forme qui garantit la portabilité des applications développées en Java.

### IV.2.5 Jena

Jena est un open source Web sémantique cadre de Java . Il fournit une API pour extraire des données et écrire sur RDF graphiques. Les graphiques sont représentés comme un «modèle» abstrait. Un modèle peut être obtenu auprès de données à partir de fichiers, bases de données, des URL ou une combinaison de ceux-ci. Un modèle peut également être interrogé par SPARQL

Fournit un environnement de développement sémantique pour RDF, RDFS, OWL et les requêtes SQL

## IV.3 NOTRE IMPLEMENTATION :

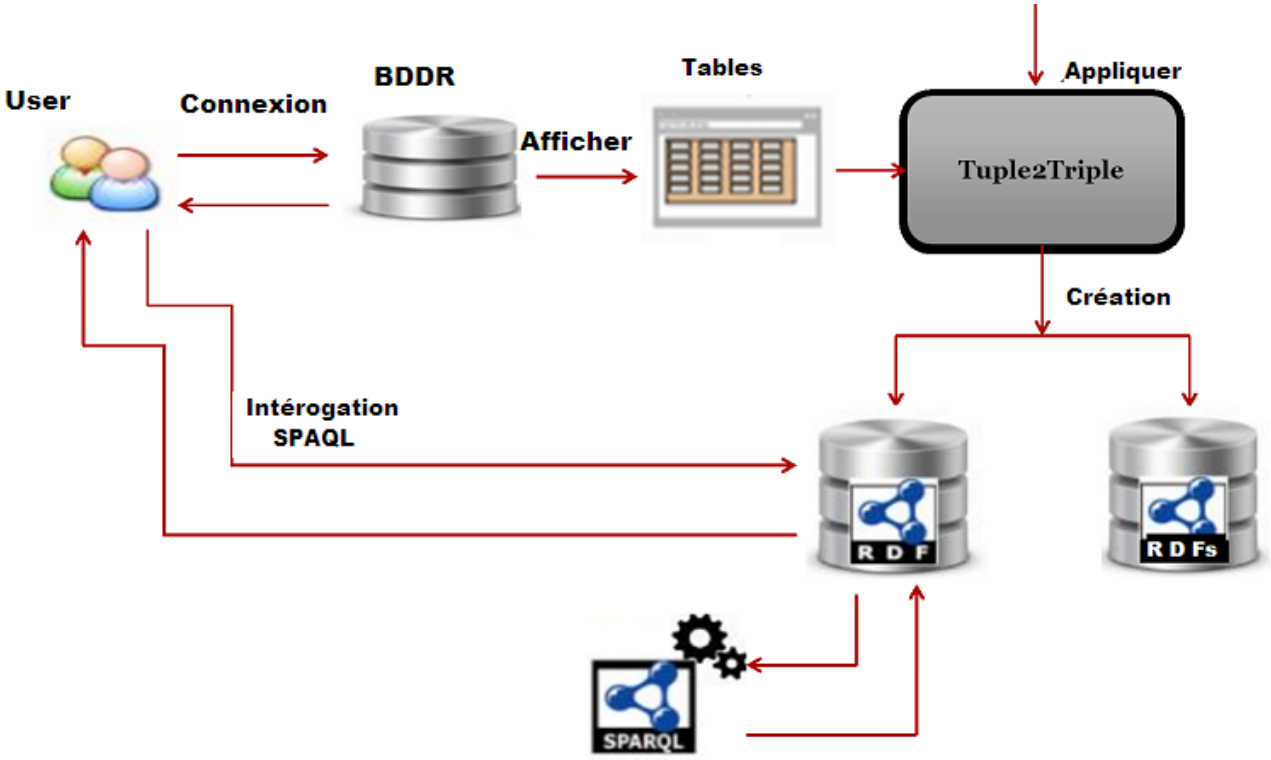


**Figure 16 : Architecture générale de l'application**

Avant tout, le développement de l'application commence par la connexion avec la base de données. Celle-ci peut toujours être modifiée dans le sens où on peut connecté avec d'autres BDD, ajouter d'autres tables ou d'autres champs de table, ou en supprimer, ou modifier ceux déjà existants.

L'utilisateur peut connecter à une BDD via l'interface de l'application, une fois la connexion est établis, les tables de la BDD choisi seront affichées. Quand l'utilisateur clique sur transformer la BDD choisi, le module Tuple2Triple effectue une transformation automatique ce qui génère en

conséquence les deux fichiers RDF et RDFS. Une interrogation peut être effectuée sur les données RDF générés à travers l'interface SPARQL.

## VI.4 LES REGLES DE PASSAGE :

### VI .4.1 RDF :

**Règle 01 :** Le fichier RDF est un fichier XML contient une entête défini.

**Règle 02 :** Chaque enregistrement de chaque table est sous la forme XML contenant une balise ouvrante et une balise fermante composées du chemin suivi du nom de la table.

Exemple : <db :player> </ db :player>

**Règle 03 :** Chaque enregistrement d'une table est transformé en schéma RDF contenant un champ nommé « About » qui contient le chemin, suivi de la table et son identificateur.

Exemple : Ainsi l'enregistrement de la table « Player » est traduite par :

Rdf: about :&db ;player_id_1"

- La règle 3 est appliquée d'une façon récursive sur toutes les tables.

**Règle 04 :** Chaque champ de l'enregistrement est défini en triplet contenant un sujet, un prédicat, et un objet.

Le sujet est le nom de l'objet.

Le prédicat est le nom du champ de l'enregistrement.

L'objet est le contenu du champ.

- Les champs de cette règle sont des attributs de la règle 2.
- La règle 4 est appliquée d'une façon récursive sur tous les enregistrements de toutes les tables.

**Règle 05 :** Si la table contient des clés étrangères, elle sera transformée en attribut nommé « Ressource » qui contient le chemin de la table et l'enregistrement a qui appartient.

**Exemple :**

Book

| ID | Author | Title | Publisher | Year |
|---|---|---|---|---|
| ISBN 0-00-6511409-X | id_xyz | The Glass Palace | id_qpr | 2000 |

Author

| ID | Name | Homepage |
|---|---|---|
| id_xyz | Ghosh, Amitav | http://www.amitavghosh.com |

Publisher

| ID | Publisher's name | City |
|---|---|---|
| id_qpr | Harper Collins | London |

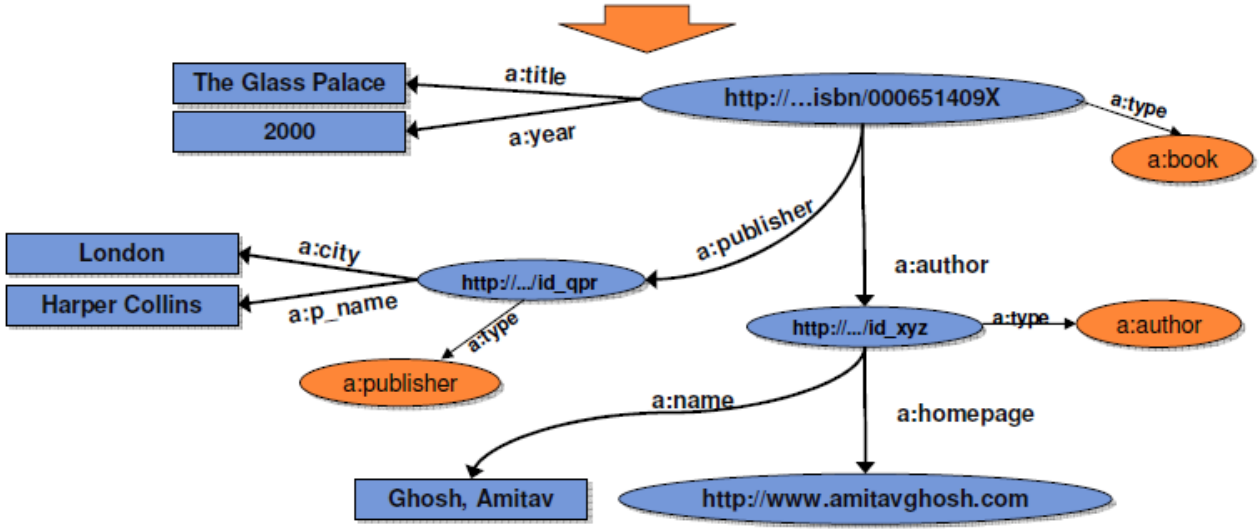


**Figure 17 :** Exemple de Transformation BDDR en RDF

Chaque table est mappée en type objet.

Tous les attributs sont transformés en prédicat.

Chaque enregistrement devient objet.

Seul les identificateurs devient des URI.

### VI .4.2 RDFs :

**Règle 01 :** Le fichier RDFs est un fichier XML contient une entête défini.

**Règle 02 :** Chaque table est définie dans RDFs comme étant une classe sous un format XML avec une balise ouvrante et une balise fermente.

<rdfs : Class> </rdfs : Class>

**Règle 03 :** La classe du schéma RDFs est composé d'un attribut nommé "About" qui contient le chemin suivi du nom de la table.

Un attribut ***Label*** qui contient le nom de la table.

**Règle 04 :** En schéma RDFs, la classe contient un élément nommé "Ressource".

**Règle 05 :** Chaque colonne d'une table est transformée en élément nommé "Property" ayant une balise ouvrante et une balise fermente.

Elle contient un attribut nommé "About". Et un autre attribut "Label".

**Règle 06 :** Chaque élément "Property" contient des sous éléments nommés "Domaine" et "Range".

**Règle 07 :** Si la table contient une clé étrangère, l'attribut "Ressource", l'élément "Range" contient le nom de la table suivi du nom de clé étrangère. Sinon, elle contient "Littéral".

## VI.5 ALGORITHME DE GENERATION DE FICHIER RDF :

**Entrées** : une BDD relationnelle <EE> ou EE est l'ensemble de l'enregistrement.

**Sorties :** fichier RDF.

**Pour RDF faire** :

Pour chaque table faire :

- Créer un élément nom de la table.
- Créer un attribut "About" prend comme valeur chemin, nom de la table concaténé avec identificateur avec le contenu.

Pour chaque EE :

- Créer tous les attributs qui contiennent comme valeur leur contenu.

Si EE content une clé étrangère

- Créer un attribut "Ressource" qui contient comme valeur le chemin suivi du nom de la table concaténé avec l'identifiant de la ressource

Finsi

Fin pour EE

Fin pour table.

## VI.6 ALGORITHME DEGENERATION DE FICHIER RDFs :

**Entrées** : une BDD relationnelle <T, C1, Co> ou T est l'ensemble des tables, Cl l'ensemble des classe et Co est l'ensemble des colonnes.

**Sorties :** fichier RDFs.

**Pour RDFs faire :**

Pour chaque T :

- Créer un élément Cl.

L'élément Cl contient un attribut "About", un attribut "Label", et un sous élément "SubClassOf"

Pour chaque Co

- Créer l'élément "Property".

L'élément Property contient un attribut "About", "Label" et des sous éléments "domaine" et "Range".

Si EE contient une clé étrangère alors le sous élément "Range" contient un attribut "Ressource" qui contient le chemin, le nom de la table suivi du nom de la clé étrangère.

Sinon "Range" prend la valeur "Littéral".

Finsi

Fin pour Co

Fin pour table

## IV.7 NOTRE APPLICATION :

Sur la fenêtre principale le client se connecte au serveur

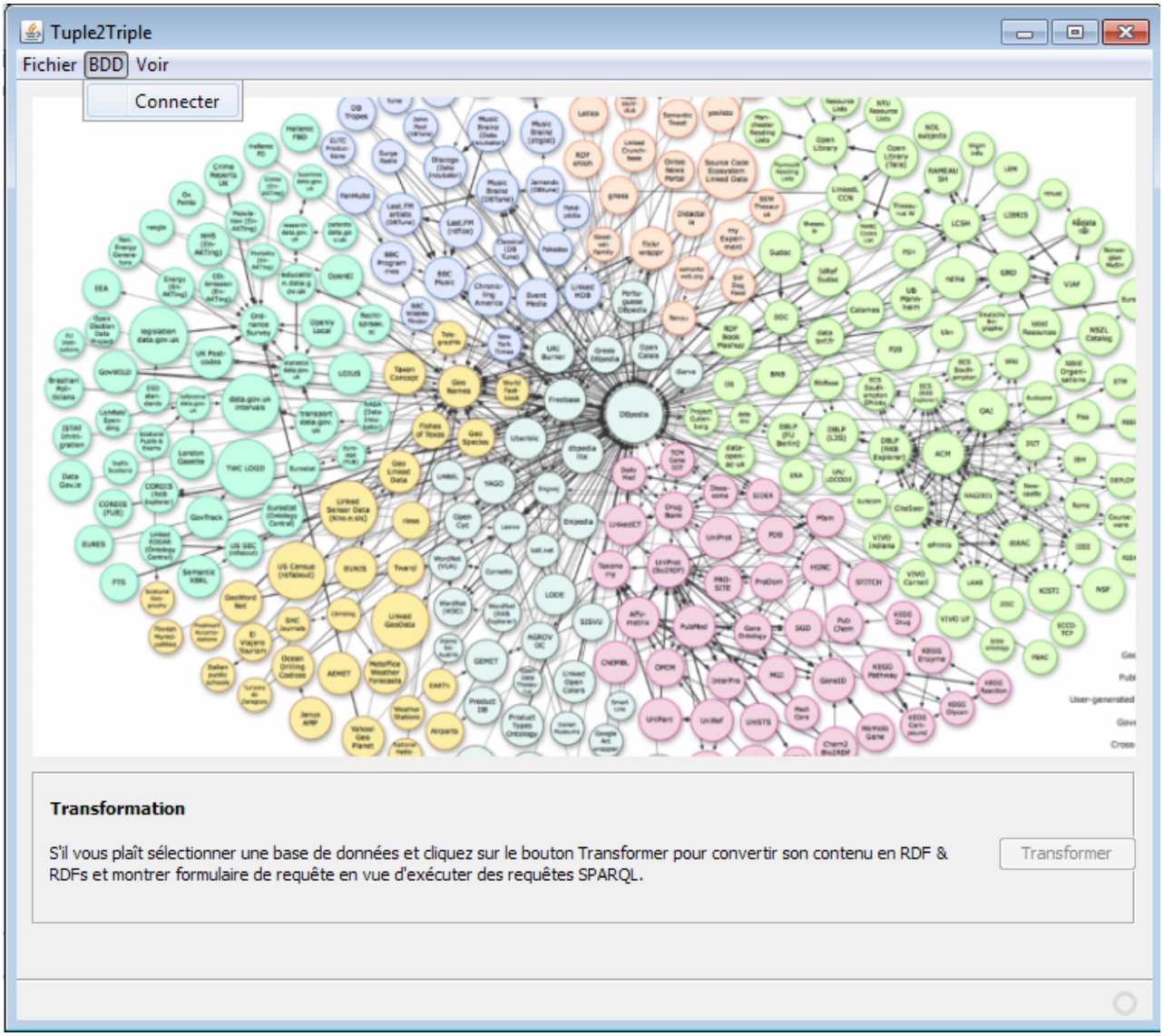


**Figure 18 :** Fenêtre principale

La troisième étape consiste à choisir une base de donnée, et dans cette étape qu'ont visualise les tables avec les attribue, Type, la Clé primaire et les Clés étrangères.

Et on suite les fichiers générer tel que le fichier RDF :

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
      <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
      <!ENTITY db 'http://db2rdf.org/db#'>
      <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
      xmlns:db="&db;"
      xmlns:rdfs="&rdfs;">
<db:Player rdf:about="&db;Player_id_1"
      db:player_id="1"
      db:player_familyname="Djokovic"
      db:player_givenname="Novak"
      db:player_birthdate="1987-05-22"
      db:player_count="4"
      db:player_size="188"
      db:player_weight="80"
      db:player_livingcity="Belgrade"
      db:player_start="2003"
      db:player_game="droitier, revers Ã  deux mains"
      db:player_ranking="2"
      db:player_points="0"
      >
</db:Player>
<db:Player rdf:about="&db;Player_id_2"
      db:player_id="2"
      db:player_familyname="Querrey"
      db:player_givenname="Sam"
      db:player_birthdate="1987-11-07"
      db:player_count="0"
```

**Figure 19 :** Fichier RDF générer

Et le fichier RDFS générer :

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
      <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
      <!ENTITY db 'http://db2rdf.org/db#'>
      <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
      xmlns:db="&db;"
      xmlns:rdfs="&rdfs;">
<rdfs:Class rdf:about="&db;Player"
      rdfs:label="Player">
      <rdfs:subClassOf rdf:resource="&rdfs;Resource"/>
</rdfs:Class>
<rdf:Property rdf:about="&db;player_id"
      rdfs:label="player_id">
      <rdfs:domain rdf:resource="&db;Player"/>
      <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&db;player_familyname"
      rdfs:label="player_familyname">
      <rdfs:domain rdf:resource="&db;Player"/>
      <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&db;player_givenname"
      rdfs:label="player_givenname">
      <rdfs:domain rdf:resource="&db;Player"/>
      <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&db;player_birthdate"
      rdfs:label="player_birthdate">
```

**Figure 20 :** Fichier RDFS générer

**Insuffisances de RDF :** RDF permet de représenter des déclarations de propriétés sur des ressources...

Mais ne permet pas d'exprimer des connaissances sur les propriétés ou sur les types de ressources :

Quelles sont les propriétés autorisées sur un type de ressources ?

Quelles sont les valeurs autorisées pour une propriété ?

Quels sont les liens entre les types de ressources (généralisation / spécialisation) ?

RDF-Schéma est un « système de typage » pour RDF, comparable à l'approche orientée objet qui permet de décrire des classes et des propriétés [21].

## VI.7 Conclusion :

Dans ce mémoire, nous avons présenté notre travail concernant la transformation de base de données relationnel en RDF. Nous avons commencé ce voyage en rappelant les différents concepts

nécessaires pour comprendre clairement le problème à savoir la définition du Web sémantique, une brève présentation des URI, le Resource Description Framework et Linked Data.

Pour répondre au différent besoins de ces cas d'utilisation, nous avons compilé un état de l'art concernant les le problème. Dans cet état de l'art, nous avons présenté les principales solutions telles que OpenLink Virtuoso Universal Server, serveur D2R, db2triples et Triplify et les deux sous standards développement par le W3C, Direct Mapping (DM) et relationnelle à RDF cartographie Langue (R2RML).

Nous avons conclu cet état de l'art en comparant les diverses forces et faiblesses de chacun de ses composants mentionnés ci-dessus.

Ensuite on a montré les règles de passages et les algorithmes utilisés pour faire la transformation.

# *CONCLUSION GENERALE*

Le succès du web de données présuppose une capacité à produire des ressources interconnectées, en grande quantité, et dotées d'une sémantique explicite les rendant directement interprétables par des applications tierces.

Pour répondre à des exigences d'ordre aussi bien qualitatif que quantitatif, l'exploitation de bases de données existantes pour alimenter le web de données est une orientation qui s'est très vite imposée et qui a ces dernières années suscité des développements technologiques significatifs.

La conversion de données en RDF constitue une étape préliminaire dans le processus de publication de données sur le Web. Plusieurs solutions ont été proposées pour accomplir cette tâche. Leur rôle principal est de transformer (semi)automatiquement des bases de données relationnelles en RDF.

D'autres outils d'interface complets existent pour éditer à la fois les données RDF et leur schéma RDFS.

Dans le présent travail nous avons proposés un ensemble de règles permettant de produire aisément des données en RDF et leur schéma RDFS.

L'approche proposée est supporté par un outil nommé *Tuple2Triple*. A travers cet outil on peut charger une base de données et de générer automatiquement des données en format RDF et leur schéma RDFS.

L'outil Tuple2Triple fournit une interface d'interrogations de données RDF à partir des requêtes écrites en SPARQL.

**Parmi les travaux futurs que nous envisageons :**

Améliorer l'approche proposé pour quel puisse générer le schéma OWL.

La phase de liage de données n'est pas encore prise en charge par notre outil, pour cela nous envisageons de l'étendre par un module assurant cette tache ainsi les données résultantes peuvent être liées à d'autre données à fin de promouvoir l'interopérabilité des jeux de données ou encore intégrer les données dans le LOD Cloud.

# *Acronymes et abréviations*

**SPARQL** Simple Protocol and RDF Query Language

**RDF** Ressource Description Framework

**RDFS** RDF Schema

**URI** Uniform Resource Identifier

**URL** Uniform Resource Locator

**W3C** World Wide Web Consortium

# *BIBLIOGRAPHIE*

**[1]** C.Bizer, T. Heath et T. Berners-Lee. Linked data - the story so far. Int. J. Semantic Web Inf. Syst., vol. 5, no. 3, page 1–22, 2009.

**[2]** T.Berners-Lee, J. Hendler, O. Lassila, The semanticweb. Sci Am 284(5):34–43, 2001.

**[3]** S. Boutemedjet , Web Sémantique et e-Learning, 2004.

**[4]** B.Ravet. MEMOIRE pour obtenir le Titre professionnel "Chef de projet en ingénierie documentaire" INTD, novembre 2011, De l'usage du Web de données pour une recherche efficace sur des ressources disséminées et hétérogènes

**[5]** T. Berners-Lee. Linked-data design issues. W3C design issue document (on-ine),http://www.w3.org/DesignIssues/LinkedData.html,06 2009.

**[6]** A. Caron, Master MIAGE/IPI-NT, 2013-2014.

**[7]** S.Degueldre, Formation des Linked Open Government Data Travail pour le cours d'Architecture des Systèmes d'Information, Décembre 2011.

**[8]** H. Fadili, Problématiques d'usage et d'intégration des langues peu dotées dans le Web des données ouvertes (Linked Open Data ou LOD) : Cas de l'Amazighe ,2011.

**[9]** V. Pasquier, MEM0R1ES - Reclaim your digital life Projet d'approfondissement, 2013.

**[10]** I. TIDDI, Découverte de patrons de dépendances pour la construction d'ontologies, septembre 2012.

**[11]** G. Gardarin, Bases de données, editions Eyrolles, 2003.

**[12]** A. Cornuéjols, bases de données concepts et programmation, AgroParisTech, Spécialité Informatique 2009-2010.

**[13]** O. Lassila, R. Swick, Resource Description Framework (RDF) Model and Syntax Specification, W3C Recommendation, 22 February 1999.

**[14]** K. Pfeifer, Semantic ETL from structured data sources v.2, October 2012.

**[15]** E. Dzalé, Y. Kaboré, Publication de données d'observation dans le Web de données, 2013.

**[16]** Virtuoso On-line Tutorials and Demonstrations http://demo.openlinksw.com:/tutorial/

**[17]** N. Bugnon, Bases de données en sciences humaines Création et pérennisation, septembre 2013.

**[18]** O. Boussaid, Le processus d'ETL, (Data Warehouses), 2013-2014.

**[19]** http://tomheath.com/papers/bizer-heath-berners-lee-ijswis-linked-data.pdf,Freie Universität Berlin, Germany, Talis Information Ltd, United Kingdom, Massachusetts Institute of Technology, USA, Juillet 2011 (Consulté le 26/02/2013).

**[20]** http://www.techno-science.net/

**[21]** http://www.info.univ-angers.fr/pub/genest/fichiers/m1_ws/ws_chap4.pdf

# Résumé :

Très nombreuses données disponibles actuellement sur le Web qui pourraient contribués au Web de donnée ce qui facilite leur accès, partage et alignement. Malheureusement, la majorité de ces données contenues dans les bases de données restent inaccessible par les applications Web. Le W3C a proposé des standards de représentation des données et leur schéma (RDF et RDFS).

Dans le présent travail nous proposons un ensemble de règles de passage qui prend au départ des données d'une base de données et génère d'une manière automatique des triples RDF et leur schéma RDFS.



**Mots –clés** : Linked Data, RDF, RDFS, web de données, bases de données, SPARQL.



# Abstract:

Many data currently available on the web that could contribute to the Web data which facilitates access, sharing and alignment. Unfortunately, the majority of these data in the databases are not accessible by web applications. The W3C proposed standard data representation and schema (RDF and RDFS).

In this work we propose a set of rules of passage that takes data out of a database and generates automatically a triple RDF and RDF schema.



**Keywords:** Linked Data, RDF, RDFS, web data, databases, SPARQL.



# ملخص:

البيانات الواسعة المتاحة حاليا على شبكة الإنترنت التي تساهم في بيانات الإنترنت مما يسهل الوصول والمشاركة والتوافق. ولكن لسوء الحظ، معظم هذه البيانات الموجودة في قواعد البيانات تبقى يتعذر الوصول إليها من قبل تطبيقات الانترنت. تقترح W3C معايير لتمثيل البيانات ومخططاتها (RDF et RDFS).

نقترح في هذا العمل مجموعة قواعد مرور التي تأخذ البيانات من قاعدة بيانات ويولد تلقائيا من الثلاثيRDF ومن مخططاته RDFS.



**الكلمات المفتاحية**: RDF، RDFS، بيانات الانترنت، قواعد البيانات، SPARQL، البيانات المرتبطة.