A Review of Accessing Big Data with Significant Ontologies

Jumah Y.J Sleeman [1, *], Jehad A.H Hammad [2]

Department of Computer Information Systems, Al-Quds Open University,
Beit Jalla, The Main road-Khallat Al Badd, Bethlehem, Palestine
[1] jsulaiman@qou.edu *;* [2] jhammad@qou.edu;
* corresponding author

ABSTRACT

Ontology Based Data Access (OBDA) is a recently proposed approach which is able to provide a conceptual view on relational data sources. It addresses the problem of the direct access to big data through providing end-users with an ontology that goes between users and sources in which the ontology is connected to the data via mappings. We introduced the languages used to represent the ontologies and the mapping assertions technique that derived the query answering from sources. Query answering is divided into two steps: (i) Ontology rewriting, in which the query is rewritten with respect to the ontology into new query; (ii) mapping rewriting the query that obtained from previous step reformulating it over the data sources using mapping assertions. In this survey, we aim to study the earlier works done by other researchers in the fields of ontology, mapping and query answering over data sources.

I. Introduction

Accessing and managing information in the big data scenarios is extremely difficult due to the multi dimensions of big data: (1) Volume which cares about the size of the data, especially the non-traditional data which produce terabytes of data within minutes. (2) Variety that represent the data stream such as social media. (3) Velocity which refers to the data types. (4) Value that refers to the valuable information that is hidden in non-traditiona1 data.

Ontology-based data access (OBDA) is a promising paradigm for solving the problem of accessing these massive amounts of accumulated data and to designing effective platforms for accessing data [1]. Figure 1 represents OBDA characteristic that consists of: 1) An ontology that represents a conceptual view of the data for a domain of interest. 2) Mapping layer that is able to solve the problems arising from the difference between the basic elements managed by data sources and the elements managed by the ontology. 3) The data sources are the repositories used in the organizations by different services and applications [1][2][3][4]. Thus, OBDA system behaves as a form of information integration that replace the global schema with a general ontology-based and end user oriented query interface over diverse data sources. Ontology with the corresponding mappings to the data sources are offering the required documentations for collecting the correct data to be returned to the client.

OBDA specifications focus on the role of answering queries to insure that they give the same answers to the considered queries for all possible extensions of data sources [4]. The life cycle of OBDA system starts from the point that end-users pass their SPARQL queries over a visual interface to the ontology layer without any knowledge of the actual structure of the data. Ontology rewrites the query obtained using one of the description logic notations that exists behind ontology. The previous query is rewritten again with respect of a mapping assertions over the data sources to get the query answer. In this scenario end-users and experts can access big data without asking IT experts.
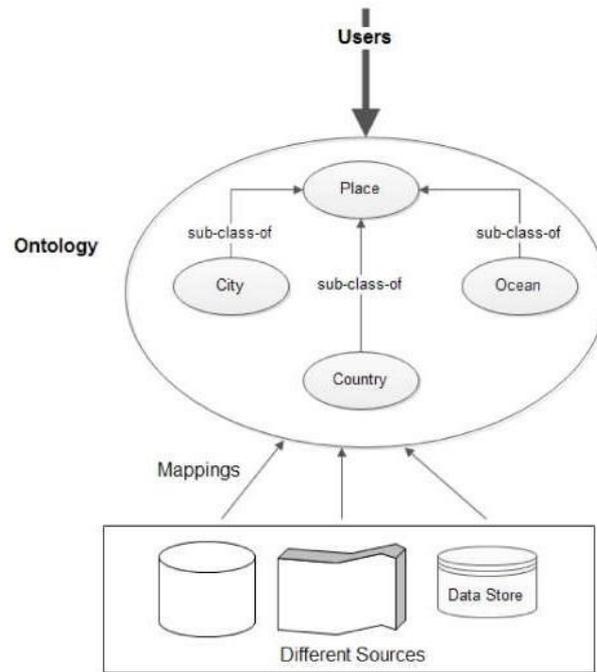
Fig. 1. OBDA Characteristic

To make this idea clearer, let us assume that the ontology T is given by a set of semantics represented by description logic's (DL$_s$). D is a relational database compatible with data sources S, and M is the mapping assertions each one of the from, $\phi(\vec{x}) \rightarrow \psi(\vec{x})$ where $\phi(\vec{x})$ is a query over S that returning rows of values for $\vec{x}$, and $\psi(\vec{x})$ is a query over T whose free variables are from $\vec{x}$ [2]. Later in this paper review, we will see how ontology, mappings as inputs, can help end-users compute a query that can be executed over the data sources

II. Motivation

In the uniform sources of data, the execution time for queries can be retrieved within minutes or seconds in the different sources. End-users need to collaborate with some IT skilled experts to develop queries that retrieve the required data. In this scenario the time round between asking and retrieving the results may be in the range of days or more. So the challenge here, is how end-users and experts can access big data without asking IT experts.

OBDA system is a recently proposed approach to address the problem of the direct access to data. It is integrated from several sources to avoid the bottleneck by automating query translation process, OBDA can be considered as a virtual approach which tells us where the exact direction of data is. OBDA also solves the problem of structural heterogeneity in which different information systems store their data in different structures and semantic heterogeneity which refers to the content of information items and its intended meanings [5].

There are several features for a successful implementation for OBDA that lead us to believe it is the right approach for end-users to access Big Data [2][4][5]:

- Ontologies: The objective of an ontology on OBDA system is to describe the domain, classifying and categorizing the elements contained within it.
- Mapping Assertion: Ontology plays an important role in information integration; it puts together all information of different formatting. In order to support data integration, mapping connect ontologies with data sources.
- Query Answering: The database queries used in OBDA are typically conjunctive queries in first-order-logic. These queries can be categorized into two: (i) Instance queries (IQ$_s$) that ask for the instance of a single concept between OBDA specifications. (ii) Union of conjunctive queries (UCQ$_s$) that ask for a set of queries between OBDA specifications

In order for end-users to create value of the data which rapidly increase, OBDA also considered the following points: (1) it is declarative, therefore no need for end-users and IT experts, to write special purpose program code. (2) Relational databases can remain as they are, hence no need for moving large and complex data sets. (3) OBDA is an adaptive system according to data scalability so data retrieving remains stable. (4) OBDA hide the complicity of data sources for the end-users. (5) The relationship between the ontology concepts and the data sources, provides a means for the experts (database administrators) to make their knowledge available to the end-user.

III. Problem Statements

A. Data Sources and Big Data

Data sources can be designated as structured or unstructured data. The term structured data refers to an identifiable structure in which the data is stored based on a methodology of columns and rows; also it is organized for human readers in a way that the data is becoming searchable by its types within content. The term "unstructured data" refers to any data that has no identifiable structure such as videos, emails, documents and texts, each of which has its own structure or format.

Big data is an expression that refers to a collection of enormous and complex data sets being generated and accumulated through three levels: the employees in companies who enter the data into the computer systems, the users who could generate the wrong data through signing up into websites such as Facebook; this level is larger than the first one according to the magnitude, and thirdly the accumulated data are derived from several machines (Satellites, sensors, robots, *etc*.).All the three levels, produce together the big data which have three main characteristics: volume, velocity, and variety. However, [6] adds one more characteristic: value; the justification is that there is a lot of information hidden in larger bodies of nontraditiona1 data so the challenge is to identify what is valuable, and then transform and extract the relevant data for analysis [7].

B. Ontology Rules

Ontologies are the structural frameworks for organizing information represented in a formal definition of the types, properties and interrelationships of the entities that exist in some domain. However ontologies take over additional tasks as discussed in following sections.

1) Content Explication

Sing1e ontology approaches [2][5][8] in Figure 2 single global ontology provide a shared vocabulary, such that all information sources are related to one global ontology and mapped to local data sources for information retrieval. This approach is not effective if one information source has a different view on a domain in addition to its sensitivity to the changes in information sources, any changing imply changes in global ontology and mapping data source.

Multi ontology approaches [2][5][8] in Figure 3: 1) Each information source is described by its own ontology. 2) Each source ontology can be developed without respect to other sources or their ontologyies. 3) It can simplify the integration task. 4) Not effective in comparing different source ontologies due to the lack of a common vocabulary.

Hybrid ontology approaches [2][5][8] these ontologies are built from a global shared vocabulary to make them comparable. In Figure 4: 1) Semantic of each source is described by its own ontology. 2) No need for modifications in mapping or shared vocabulary in terms of adding new sources. 3) It is extremely hard to reused existing ontologies because all sources refer to the shared vocabulary.
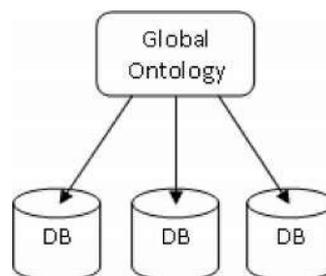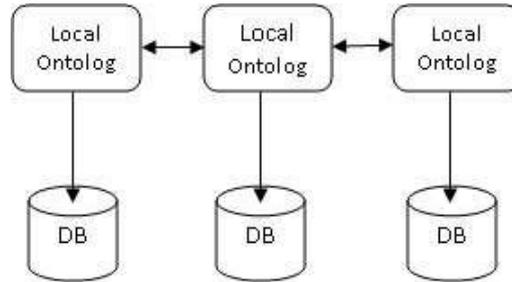

Fig. 2. OBDA characteristic
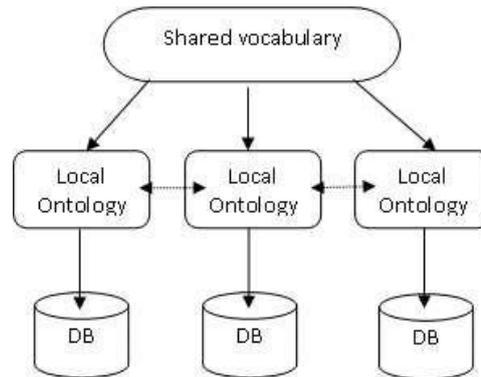
Fig. 3. Multiple ontology approach

Fig. 4. Hybrid ontology approach

2) Ontology knowledge

Description logic's are logic's specifically designed to represent the structured knowledge to represent a domain that composed of objects and structured into: (i) Concepts which correspond to a classes and denote sets of objects. (ii) Roles which correspond to (a binary) relationships and denote binary relations on objects.

Web Ontology Language (OWL) is a richer vocabulary description language for describing properties and classes. The formal underpinning of OWL is based on Description Logic's (DLs) knowledge representation formalisms with well-understood computational properties [9]. DL ontology consist of the Terminological Box (TBox) and Assertion Box (ABox), Tbox describe a system in terms of controlled vocabulary such as a set of classes and properties. ABox is a TBox statements that represents the ontology vocabu1ary, TBox and ABox together representing the base knowledge (KB).

DLs are a family of logic's concerned with knowledge representation, it is a decidable fragment of first-order-logic (FOL) associated with a set of automatic reasoning procedures. The basic constructs of a DL are the notion of a concepts and the notion of relationship. Complex concept and relationship expression can be constructed from atomic concepts and relationships with suitable constructs between them [4][9]. Since the ontology is a model of (some aspect of) the world, it can introduce vocabulary relevant to domain with specific meaning (semantics) in terms of A happy cat owner owns a cat and all cats he cares for are healthy which can be formalized using suitable description logic (DL)

$$HappyCatOwner \ \sqsubseteq \ \exists Owns.Cat \ \sqcap \ caresFor.Healthy \qquad\qquad (1)$$

The most known description logic's are [10]:

- $F\overline{L}$: The simplest and less expressive DL consist of the following concepts
  $C, D \rightarrow A \mid C \sqcap D \mid \forall R.C \mid \exists R$
- *ALC*: More practical and expressive DL consist of
  $C, D \rightarrow A \mid \top/\bot \rightarrow A \mid C \sqcap D \mid \forall R.C \mid \exists R.\top$
- *SHOIN*: A very popular DL, it's the logic underlying OWL.
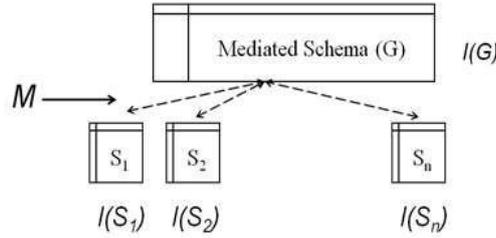- *DL - Lite$_{A,id}$ family*: A very expressive DL capable of representing most database constructs.

Fig. 5. Semantics of schema mappings

DL knowledge base ($\sum$) normally separated into two parts $\sum = $ (*TBox, ABox*), *TBox* describing the structure of a domain in the form of *C $\sqsubseteq$ D, C = D* and *ABox* a set of axioms in the form of *C(a), R(a, b)* describing the data. Further details will be found in [3][4][10][11]. Figure 5 shows the example of DL knowledge base.

TBox example
> *T = (Student = Person $\sqcap$ $\exists$Name.String $\sqcap$ $\exists$Addrees.String $\sqcap$ $\exists$Enrolled.Course Student*
> *$\sqsubseteq$$\exists$Enrolled.Course $\exists$Teacher.Course $\sqsubseteq$$\rightarrow$Undergrad $\sqcap$ Professor)*

ABox example
> *A = (Student (John) Enrolled (John, cs124) (Student $\sqcup$ Professor) (Paul))*

C. Mapping

The purpose of mapping is to reconcile heterogeneity derived from different designed schema's even if the people or organizations are model the same domain, mostly these problems happened between the mediated schema and the schema of the data sources. In Figure 5, schema mappings describe the relation in which instances of the mediated schema are consistent with current instances of the data sources [12]. *I(G)(I(S$_i$))*: the set of possible instances of the mediates schema G(S).

$$M_R \subseteq I(G) \times I(S_1) \times I(S_2) \times ... I(S_n) \tag{2}$$

Mapping M_R represent all possible instances of mediated schema *G* given instances of sources $I(S_1)$, $I(S_2)$,... $I(S_n)$. In other words mapping assertion specifies the semantic relationship between elements of a DL TBox ontology to elements of a data sources [4].

Many OBDA studies focused on understanding which languages for the ontology and mappings allow query answering to be performed taking into account the inconsistency and redundancy for mappings OBDA [3]. Query execution can be performed if (1) the ontology is expressed in description logic DL - Lite. family ontology language, and (2) the mapping are of types (a) Globa1-as-View (GAV) in which mediated schema defined as a set of views over the data sources, in which mapping is executed from entities in the global ontology to entities in the original sources (b) Local-as-View (LAV) in which data sources defined as views over the mediated schema, in which mapping executed from entities in the original sources to the global ontology (c) GLAV the combination of the two.

Mapping analysis in OBDA aims to provide the designer with the useful services that produce a well-founded OBDA specification, thus two important points should be considered: (1) Inconsistent mapping M with respect to Ontology O and source schema S means that retrieval, data lead to inconsistent OBDA specifications even the S schema is non-empty. In other words, no data retrieval or the data are mismatched. (2) When M is subsumes of M' (M $\sqsubseteq$ M') w.r.t O and S which means that O, S and M$\cup$M' are equivalent. These proprieties are very important in the life of OBDA systems to avoid the above problems, especially when executing hundred of queries [11][13].

IV. Methodology of OBDA

In figure 6, the query that obtained from the end-user via visual query system divided into two steps: (i) Ontology rewriting, in which the query is rewritten with respect to the ontology into new query(ii) Mapping rewriting in which the query obtained is reformulated over the data sources using mapping assertions [14]. The specification of OBDA is a triple of *J = (O, S, M)* where O is the
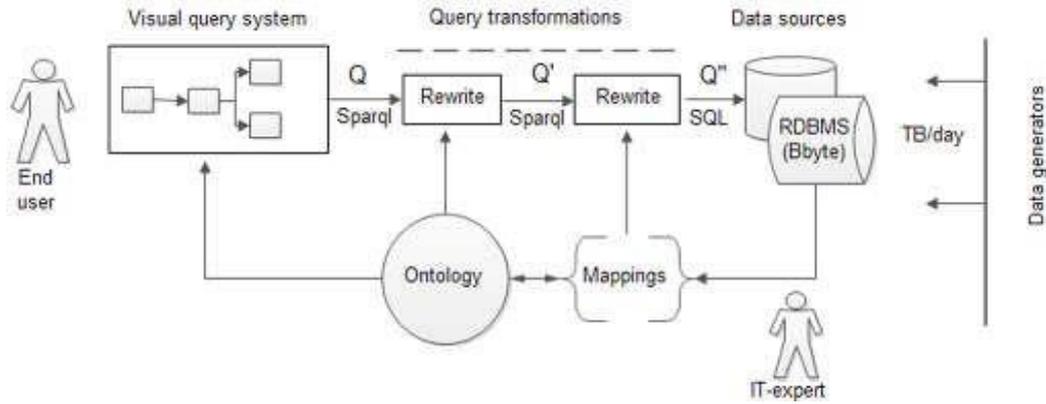
Fig. 6. OBDA query system

description logic TBox ontology, S is a source schema with integrated integrity constraints, and M is a mapping between the two consist of assertion of the form

$$\phi(x) \rightarrow \psi(x) \tag{3}$$

where $\phi(x)$ is a query over sources and $\psi(x)$ is a query over Ontology [11][13]. We donate to the O with a signature $\sum_O$ and description logic language with L_O, while S has the signature $\sum_S$ and description logic language with L_S;X is the number of arguments that the function passes. The functionality of $m \in M$ mapping assertions with the form of equation (3) [15] means:

- $\psi(x)$ query with the signature $\sum_O$ represented by head (m)
- $\phi(x)$ query with the signature $\sum_S$ represented by head (m)

where mapping assertion $m_i \in M$ and $i = \{1, 2, 3, ..\}$

With this scenario GAV mapping rewriting consists of grouping all SQL queries mapping the same ontology role to the database into a single query [3][4]. Example: A schema S of the database represented by two tables HumanTab and AreaTab for handling information about Humans and their strain, the underlined attribute represent the primary key of the table and the attribute Area represents the foreign key for the two tables

$$HumanTab(HumanCode, Name, Strain, Area)$$

$$AreaTabe(AreaCode, AreaName)$$

The ontology O is as follow

$$O = \left\{ \begin{array}{c} Africa \sqsubseteq Human, Asian \sqsubseteq Human \\ African \sqsubseteq \rightarrow Asian, \\ Human \sqsubseteq \exists Name, Human \sqsubseteq \exists Localion, \\ \exists Localion \sqsubseteq Code, Code \sqsubseteq Name \end{array} \right\}$$

In words O, specified Asian and African as Humans, Asian can not be African, and every Human has a Name and located in a Location that has a Code. Moreover, every Code has a Name. Mapping M between O and S is as follows:

$m1$: $select\ HumanCode\ as\ x, Name\ as\ y\ from\ HumanTab\ \rightarrow\ Human(x) \wedge Name(x, y)$

$m2$: $select\ HumanCode\ as\ x, Name\ as\ y\ from\ HumanTab\ where\ Strain = "African"\ \rightarrow$
$\qquad African(x) \wedge Location(x, y)$

$m3$: $select\ HumanCode\ as\ x, Name\ as\ y\ from\ HumanTab\ where\ Strain = "Asian"\ \rightarrow$
$\qquad Asian(x) \wedge Location(x, y)$

$m4$: $select\ HumanCode\ as\ x, Area\ as\ y\ from\ HumanTab\ \rightarrow\ Location(x, y)$

$m5$: $select\ AreaCode\ as\ x, AreaName\ as\ y\ from\ AreaTab\ \rightarrow\ Code(x) \wedge Name(x, y)$

The semantic of OBDA specifications j with respect of S is legal if

$$I_D \vDash S \tag{4}$$

where I_D is a set of facts over $\sum$s. In other words, for each S a legal instance, always exists. In equation (5) every mapping assertion will denote the existential arguments in the head (m) [3][4][11][13]

$$\sum_1^m \phi(x) \rightarrow \psi(x); m \in M \tag{5}$$

V. Discussions and Evaluations

A. Evaluation

The main aim of ontology rewriting query is to solve the problem of query answering that comes from the end-user. The idea behind that is to transform the given query and TBox into an expanded query that contains all relevant information captured in the TBox, also to evaluate the expanded query over ABox only. The expanded version is also formed by a union of conjunctive queries (UCQs) that avoid keeping the large ABoxes in memory [16].

Another issue is the size of the rewriting query over ontology which equal the size of TBox and the ordered query. In this case, (UCQs) will contain hundreds or thousands of queries which affect the performance of retrieving information.

Two types of problems may appear in OBDA system: (1) Syntax error, such that the ontology TBox represented by DL-Lite family semantically formulated correctly and the mapping assertion does not contain misspellings. (2) Semantic problems, where the ontology does not contain unsatisfiable concepts, roles, or attributes. The semantic problems for the mapping where a mapping assertion $m \in M$ is semantically anomalous if the answer to either the head query of m or the body query of m is empty, also of the body of the query is empty (SQL over database) then the m assertions is useless, but if the head of the query is empty (Conjunctive queries) is empty and the body is not, the assertion may lead to a contradiction [15].

B. Table Summary

In this section, we present a discussion related to the OBDA system that we present. First, we make a comparison between different systems that uses OBDA for the integration of heterogeneous information sources. We compare the ontology languages as well as to connect ontology with sources via mappings. From Table 1 we find that ontology is formulated using DL-Lite family [17][18][19][20], and DL behind OWL as shown in (1) [14][21][22][23][24]. From Table 1 most of the presented platforms used GAV mapping rewriting. Also, it shows the methodology that implement OBDA specifications and some important points that shed the light in how these systems derived the data sources.

In Table 2, we present a discussion related to mapping connection to information sources as follows: (1) Straight forward approach that connect ontology to data schema in terms of one-to-one copy of the structure of the database and encode it in a language that makes automated reasoning possible. (2) Definition approach does not correspond to the structure of the database, these are only linked to the information by the terms that is defined. (3) Structure enrichment which combine the two previously the structure and the information source. (4) Meta-Annotation that adds semantic information to an information sources which present in the World Wide Web [5].

Table 3 summarized the standard languages and the query models that we used in this review paper. GAV in which ontology is defined as a set of views over the data sources. In GLAV approach, each mapping rule is represented by a conjunctive query written in the global schema associated with a conjunctive one written in source schemas. An R2RML is a mapping language that connect the relational databases to RDF dataset throw logical tables to retrieve data from the input database. Standard languages also represent *DL - Lite* family and OWL2QL ontology languages with formally defined meaning [3][4][11][13][25]. Table 3 also shows that query answering could be a union of conjunctive query (UCQs) [3] or standalone conjunctive query (CQs) over ontology.

Table 1. Comparison between different platforms that uses OBDA, in terms of ontology language, mapping assertions, methodology and other important points

Presented Platform	Ontology Language	Mapping Assertions	Methodology	More Points	Ref.
Optique (visual query) interface	DL-Lite family	LogMap system to discover ontology-to-ontology mappings	* Every many-to-many table in BD is mapped to one class in the ontology * Every data attribute is mapped to one data prosperity * Every, foreign key mapped to one object property	Drive the ontology from database schema (reverse, Engineering)	[17] [18] [19]
SmartDairy-Farming project	TopBraid (common ontology using RDFS and OWL)	LogMap system	* Sensor, equipment for collecting data sources, * Set-up, smart ontology using SPARQL query	The project is in an experimental phase	[21]
MASTRO project using java tool	DL-Lite family behind OWL	GAV	Adding view inclusion to the OBDA specifications , which eliminate sub-queries contained into other sub-queries of rewriting queries	This study focus on the case where data is stored in ABox	[14] [22] [23]
CLIPPER system	DL-Lite family		Presenta rewriting-based algorithm for conjunctive query answering over ontologies	The experiments used TBoxs taxonomies and the quire reasoning with Horn-SHIQ (DL)	[20]
Ontop	OWL2 QL, rewriting of conjunctive queries (CQs), over ontologies (FO) queries.	Mapping, M, as a set of GAV rules	Using the mechanism of OBDA with ontologies given in OWL2 QL a profile of OWL2 designed to support rewriting of conjunctive queries (CQs), over ontologies into first-order (FO) queries.	OBDA is achievable in practice when applied to real-world ontologies, queries and data stored, in relational databases	

Table 2. Comparison in terms of mapping connection to information sources

Mapping Connection	Definition	References
Straight forward approach	Copy the structure of the database	[4][13]
Definition approach	Linked to the information by terms that is defined	[11][14][17][18][19]
Structure enrichment	Copy the structure and the information of the database	[3][9][22][23][24]
Meta-Annotation	Adds semantic information to the sources.	[20][21]

Table 3. Comparison in terms of standard languages and query model

Mapping Connection	Definition	Ref.
GAV for mapping assertion PerfectMap algorithm	Union of first-order rewritable conjunctive query (UCQs)	[3]
GAV for mapping assertion GLAV for mapping assertion TBox formulated in DL-Lite$_R$	Conjunctive query (CQs) and instance queries (IQs)	[4]
OWL2QL for ontology definition SPARQL for query specification R2RML for mapping assertion	Conjunctive query (CQs)over ontologies specified in SPARQL	[14][22][23]
DL-Lite family R2RML for mapping assertion	ABoxes turn out to be unions of lew queries whose size does not exceed the size of the original query	[17][18][19][21]
GAV for mapping assertion GLAV for mapping assertion	Conjunctive query (CQs)	[11][13]

VI. Conclusion

A promising OBDA system is able to solve many challenges related to end use of data access especially on big data. This approach presented a query answering based on two steps (i) Ontology rewriting. (ii) Mapping rewriting over data sources. A successfully OBDA implementation can solve the problem of accessing big data as follows (1) There is no need to write a special coding by the end-users or the IT experts. (2) Data can be left in the relational database. (3) It provides a flexible query language which corresponds to end-users. (4) The ontology will hide the complexity of the source schema for the end-user. (5) Database expert's knowledge will be available to end-users because of the relationship between the ontology and the sources via mapping. From this survey we have found that most of the researchers' efforts studying how to extract implicit knowledge from big data based on the use of ontologies and the declarative mappings between data and ontology schema's. Also, researchers introduced existing platforms and under constructing ones based on OBDA systems to give end users the ability to access big data through visual interfaces to write queries.

Declarations

Author contribution

All authors contributed equally as the main contributor of this paper. All authors read and approved the final paper.

Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare no conflict of interest.

Additional information

No additional information is available for this paper.

References

[1] K. T. Wassif, "A Survey on Using Ontology for Addressing End User Access to Big Data," *Int. J. Comput. Syst.*, vol. 02, no. 08, pp. 363–372, 2015.

[2] M. Giese *et al.*, "Scalable End-User Access to Big Data," in *Big Data Computing*, Chapman and Hall/CRC, 2013, pp. 205–244.

[3] F. Di Pinto *et al.*, "Optimizing query rewriting in ontology-based data access," *Proc. 16th Int. Conf. Ext. Database Tech. - EDBT '13*, 2013, p. 561.

[4] M. Bienvenu and R. Rosati, "Query-based comparison of OBDA specifications," *Proc. 28th Int. Work. Descr. Logics (DL 2015)*, 2015.

[5] H. Wache *et al.*, "Ontology-based Integration of Information - A Survey of Existing Approaches," *IfCAI-01 Work. Ontol. Inf. Shar.*, pp. 108–117, 2001.

[6] J. P. Dijcks, *Oracle: Big data for the enter- prise*. Oracle White Paper, 2012.

[7] S. . Jeong and I. Ghani, "Semantic Computing for Big Data: Approaches, Tools, and Emerging Directions (2011-2014)," *KSII Trans. Internet Inf. Syst.*, vol. 8, no. 6, pp. 2022–2042, Jun. 2014.

[8] L. Zuo, "A semantic and agent-based approach to support information retrieval, interoperabil- ity and multi-lateral viewpoints for heterogeneous," University of London, 2006.

[9] B. C. Grau, E. Kharlamov, and D. Zheleznyakov, "How to Contract Ontologies - Statement of Interest," pp. 1–5, 2012.

[10] M. Jarrat, "Towards methodological principles for ontology engineering," Llniversiteit Brassel, 2005.

[11] D. Lembo, J. Mora, R. Rosati, D. F. Savo, and E. Thorstensen, "Mapping Analysis in Ontology-Based Data Access: Algorithms and Complexity," *Lecture Notes in Computer Science*, pp. 217–234, 2015.

[12] V. Jahns, "Principles of data integration by Anhai Doan, Alon Halevy, Zachary Ives," *ACM SIGSOFT Softw. Eng. Notes*, vol. 37, no. 5, pp. 43–43, Sep. 2012.

[13] D. Lembo, J. Mora, R. Rosati, D. F. Savo, and E. Thorstensen, "Towards Mapping Analysis in Ontology-Based Data Access," *Conf. Int. Conf. Web Reason. Rule Syst.*, pp. 108–123, 2014.

[14] N. Antonioli *et al.*, "Ontology-based data access: The experience at the Italian Department of Treasury," *CEUR Workshop Proc.*, vol. 1017, no. January, pp. 9–16, 2013.

[15] D. Lembo, R. Rosati, M. Ruzzi, D. F. Savo, and E. Tocci, "Visualization and management of mappings in ontology-based data access (progress report)," *CEUR Workshop Proc.*, vol. 1193, no. October, pp. 595–607, 2014.

[16] H. Pérez-Urbina, E. Rodríguez-Díaz, M. Grove, G. Konstantinidis, and E. Sirin, "Evaluation of query rewriting approaches for OWL 2," *CEUR Workshop Proc.*, vol. 943, pp. 32–44, 2012.

[17] M. Giese *et al.*, "Optique: Zooming in on Big Data," *Computer (Long. Beach. Calif).*, vol. 48, no. 3, pp. 60–67, Mar.

2015.

[18] D. Calvanese *et al.*, "The optique project: Towards OBDA systems for industry," *CEUR Workshop Proc.*, vol. 1080, no. January 2015, 2013.

[19] D. Calvanese *et al.*, "Optique: OBDA Solution for Big Data," *Semant. Web ESWC 2013 Satell. Ereitfs. Springer*, pp. 293–295, 2013.

[20] T. Eiter, M. Ortiz, M. Šimkus, T. K. Tran, and G. Xiao, "Query rewriting for Horn-SHIQ plus rules," *Proc. Natl. Conf. Artif. Intell.*, vol. 1, no. c, pp. 726–733, 2012.

[21] J. P. C. Verhoosel, M. Van Bekkum, and F. K. Van Evert, "Ontology matching for big data applications in the smart dairy farming domain," *CEUR Workshop Proc.*, vol. 1545, pp. 55–59, 2015.

[22] D. F. Savo *et al.*, "Mastro at work: Experiences on ontology-based data access," *CEUR Workshop Proc.*, vol. 573, no. June 2014, pp. 20–31, 2010.

[23] D. Calvanese *et al.*, "The MASTRO system for ontology-based data access," *Semant. Web*, vol. 2, no. 1, pp. 43–53, 2011.

[24] R. Kontchakov, M. Rodríguez-Muro, and M. Zakharyaschev, "Ontology-Based Data Access with Databases: A Short Course," *Lect. Notes Comp. Sci.*, 2013, pp. 194–229.

[25] L. E. T. Neto, V. M. P. Vidal, M. A. Casanova, and J. M. Monteiro, "R2RML by Assertion: A Semi-automatic Tool for Generating Customised R2RML Mappings," *Lect. Notes Comp. Sci.,* 2013, pp. 248–252.