The use of ontologies for effective knowledge modelling and information retrieval

CrossMark

Kamran Munir [a,*], M. Sheraz Anjum [b]

[a] Department of Computer Science and Creative Technologies, University of the West of England, BS16 1QY Bristol, United Kingdom
[b] School of Electrical Engineering and Computer Science, National University of Sciences and Technology (NUST), H-12 Islamabad, Pakistan

ARTICLE INFO

ABSTRACT

The dramatic increase in the use of knowledge discovery applications requires end users to write complex database search requests to retrieve information. Such users are not only expected to grasp the structural complexity of complex databases but also the semantic relationships between data stored in databases. In order to overcome such difficulties, researchers have been focusing on knowledge representation and interactive query generation through ontologies, with particular emphasis on improving the interface between data and search requests in order to bring the result sets closer to users research requirements. This paper discusses ontology-based information retrieval approaches and techniques by taking into consideration the aspects of ontology modelling, processing and the translation of ontological knowledge into database search requests. It also extensively compares the existing ontology-to-database transformation and mapping approaches in terms of loss of data and semantics, structural mapping and domain knowledge applicability. The research outcomes, recommendations and future challenges presented in this paper can bridge the gap between ontology and relational models to generate precise search requests using ontologies. Moreover, the comparison presented between various ontology-based information retrieval, database-to-ontology transformations and ontology-to-database mappings approaches provides a reference for enhancing the searching capabilities of massively loaded information management systems.

Contents

* Corresponding author.
   E-mail address: Kamran2.Munir@uwe.ac.uk (K. Munir).

1. Introduction

In information management systems, structured query formulation languages are one means of retrieving information. Writing structured queries is a powerful method to access data since it allows end-users to formulate complex database queries by learning specialised query languages. However, query formulation with the exception of a few visual query generation and refinement approaches remains appreciatively difficult for the various levels of systems users. In recent years information retrieval has turned out to be more complicated with the increased use of data mining, decision support and business analytics applications. Consequently, researchers focus has been on approaches that include *visual database interfaces* [1] and interactive query generation through graphs [2,3], with a particular emphasis on providing *interactive natural language* interfaces to support query generation. Recently, *semantic-based* approaches using domain ontologies have been adapted for data modelling and information retrieval. *Ontology-based information retrieval*, for example as in [4–6] mainly aim at improving the interface between data and search requests in order to bring the result sets closer to the users' research requirements. In general, an ontology represents a shared, agreed and detailed model (or set of concepts) of a certain problem domain [7]. One major advantage of using a domain ontology is its ability to define a semantic model of the data combined with the associated domain knowledge. Ontologies can also be used to define links between different types of semantic knowledge. Thus, ontologies can be used in formulating some data searching strategies.

This paper discusses ontology-based information retrieval approaches by taking into consideration the aspects of:

(a) *ontology generation from database schema(s)*;
(b) *processing of domain knowledge to represent it as ontological knowledge*; and
(c) *the translation of such ontological knowledge into relational database queries*.

Moreover, it provides a comparison between ontology-to-database *transformation* and *mapping* approaches in terms of: *loss of data and semantics*;

structural mapping; and
domain knowledge applicability.

The outcomes presented in this paper can be beneficial in bridging the gap between ontology and relational models while attempting to generate precise search requests from ontology expressions. Moreover, the comparison presented between various ontology-based information retrieval, database-to-ontology transformations and ontology-to-database mappings tools/approaches provides a reference for enhancing the searching capabilities of massively loaded information and management systems [8].

After having introduced the motivation and context, the remainder of this paper is divided into the following sections. Section 2 introduces ontologies and domain knowledge representations. Section 3 reviews the state of the art in ontology-based database information retrieval. Section 4 discusses our findings in relation to ontology-based information retrieval. Section 5 reviews the state of the art database schema to ontology schema transformation and ontology-to-database mapping approaches in terms of loss of data and semantics, structural mapping and domain knowledge applicability. Section 6 provides a discussion and highlights a possibility of combining database-to-ontology transformation and ontology-to-database mappings approaches for relational query formulation. Finally, Section 7 outlines the future challenges and possible research directions towards using ontologies for information retrieval from information and Big data management systems.

2. Ontologies and knowledge representation

Over the past few years, many ontology development and query languages have been developed and this is still a continuing effort. While building an ontology-based system, it first requires deciding which ontology language is to be used in a given context. Numerous ontology languages were developed in the last few years. Most of these are based on the eXtensible Markup Language (XML) [9] which enables them to be machine interpretable [10]. Notable examples are the Resource Description Framework (RDF) and RDF Schema [11], the DARPA Agent Markup Language and the Ontology Inference Layer (DAML + OIL) [12], and the Ontology Web language (OWL) [13] and OWL2 [14]. In order to use ontologies for query formulation it is important to evaluate them in terms of their expressive power, tools and reasoning support in order to decide which ontology language is best suitable for this task. Most developments in the latest ontology languages are influenced by the RDF/RDFS XML based language rules and ranking structure. OWL has three sublanguages: OWL-Lite, OWL-DL and OWL-Full; it was developed on top of RDF and DAML + OIL [15] providing gradually more expressiveness power. The Semantic Web Rule Language (SWRL) [16] adds rules to OWL-DL (to increase its expressiveness) and is a combination of the OWL-DL and sublanguages of the Rule Markup Language such as First Order Language. The basic idea of SWRL is to extend OWL-DL. It is simple and has tight integration with the existing OWL language; this can be considered as a key characteristic of SWRL. Recently, OWL2 [14] has been developed on the existing structure to OWL (OWL1) i.e. all the building blocks of OWL2 are present in OWL1; therefore all OWL1 ontologies remain valid OWL2 ontologies [14]. There are three new profiles of OWL 2: (1) OWL 2-EL, (2) OWL 2-QL, and

Table 1
Comparison between RDF(s), OWL-1 and OWL-2 showing possible uses of knowledge representation concepts to formulate ontology based relational database queries.

Concepts	RDF(s)	OWL 1	OWL 2
Formal semantics	✔	✔	✔
Equivalence	×	✔	✔
Class definitions	×	✔	✔
Constraints	×	✔	✔
Enumerations	×	✔	✔
Cardinality constraints	×	✔	✔
Inference	×	✔	✔
Property chains	×	×	✔
Disjoint properties	×	×	✔
Qualified cardinality restrictions	×	×	✔

(3) OWL 2-RL. These profiles are syntactic subsets of OWL 2 constructs. Selection between these profiles depends on the reasoning tasks and ontology structure. A comparison between RDF(s), OWL-1 and OWL-2 showing the possible uses of knowledge representation concepts to formulate ontology based relational database queries is presented in Table 1. In summary, both OWL and RDF have many common features, but OWL is a stronger language with greater machine interpretability than RDF. Moreover, OWL comes with a larger vocabulary and a stronger syntax than RDF, which can be used to define complex ontology concept restrictions and subsequently to formulate ontology based relational database queries.

3. Ontology-based information retrieval

This section reviews the state of the art in ontology-based database information retrieval. Here, a historical overview of information retrieval approaches is first presented, followed by a detailed analysis of existing ontology-based query systems and data search strategies in relation to three different key aspects that guided the review of such work. These three aspects are: (1) ontology assisted visual or interactive query formulation; (2) ontology based information linking approaches (also known as keyword search); and (3) ontology based query refinement (including query enrichment).

3.1. Information retrieval from a historical perspective

Database information retrieval is the search for information in databases. The need for effective methods to automate information retrieval has grown in importance because of the significant increase in the amount of both structured and unstructured information embodied in information sources. Over the years, many visual information retrieval approaches came into existence which aim to reduce the end users effort while interacting with databases. These approaches intend to extract information from databases using visual tools. Such approaches include form-based [17], query by example (QBE) [18] or query by template (QBT) [19] etc. These approaches work for basic relational database queries, primarily because tabular structure of the database fits well with the tabular skeletons used in query interfaces. However, such approaches do not help in semantic data retrieval nor do they provide any query formulation support to generate complex queries.

To overcome the shortcomings identified above, further implementation improvements were advocated. One example is QUICK (Universal Interface with Conceptual Knowledge) [18] that focuses on automating query formulation by exploiting ER conceptual schema design knowledge. Unfortunately, in the real world the ER model has been used primarily for database design and they often do not store domain knowledge. Therefore, the ER based query formulation approaches cannot provide a reliable method to depend on its comprehensiveness in expressing low-level query constraints. More recently, several ontology languages with properly specified semantics have been developed. Several ontology-based approaches have been reported in the literature that can provide intelligent query formulation services for relational databases. Such approaches are reviewed in the following sections.

3.2. Ontology-based query formulation approaches

Ontology-based *Visual* or *Interactive* query formulation systems are query systems for databases that use visual representations to express related data requests. These systems adapt ontologies for database query generation in order to improve the effectiveness of the human-computer communication. In recent years, many such systems have been reported in the literature (e.g., TAMBIS [20], GRQL [21], SEWASIE [22], Ontogator [23], OntoViews [24], OntoQF [25], VISAGE [26], Smartch [27], Semantic-based [28] and many others). In most of these ontology based visual query formulation systems, the search queries are performed using an ontology browser that visualises the ontology as a tree. The actual search is done via concept selection through a visual tree or through keywords annotated by the visual ontology concepts.

The TAMBIS system [20] supports the specialisation or generalisation of the base or filler ontology concepts to build database specific queries interactively. Here the data in the databases are stored (linked) as instances of ontology concepts. This approach can be applied to resolve integration problems, where all information sources have the same schema or provide nearly the same view of a domain. Another similar approach based on ontological graph pattern queries is presented in GRQL [21] and KnowledgeSifter [29]. GRQL relies on the full power of the RDF/S data model and provides a GUI for building queries based on ontology navigation. In this approach, queries are constructed by graphically navigating through individual RDF/S classes and property definitions. In SEWASIE (SEmantic Webs and AgentS in Integrated Economies) [22], the principles of designing and developing an ontology-based query interface are presented. The query interface of SEWASIE supports the user in formulating a query through an iterative refinement process supported by ontology navigation where in the query formulation process, a user can specify a request using generic terms, can refine some terms of a query or can introduce new terms, and can iterate the process if needed.

In OntoQF [25] OWL-DL ontologies have been used for information retrieval by automatically generating relational database queries using pre-stored domain knowledge. In comparison to other existing approaches, one of the main features of OntoQF approach is that it uses a combination of both database-to-ontology transformation and mappings to enable the automatic query formulation process, which helps in generating precise database queries. Overall, OntoQF uses a two-phase approach. In the first pre-processing phase, domain ontology is generated from relational schema and related mappings are defined which links the domain ontology concepts to relational entities/columns and vice versa [30]. Moreover, the domain experts can specify studies as ontology statements using a visual ontology query editor. OntoQF rules for ontology-based relational query formulation suggest that for such query formulation the generated domain ontology does not require the definition of datatype ranges [31] or specific constraints that are expressed in the database schema. Moreover, the domain knowledge is to be expressed in terms of OWL-DL assertions as concept restrictions, which need to be consistent with the respective domain ontology schema. In the second translation phase, the OntoQF engine translates ontology statements into the corresponding relational query statements. OntoQFs approach is suitable for those systems or data mining applications that aim to keep all data at the original location(s) and use domain ontology for knowledge-based information retrieval [25].

The system presented in [32] provides interactive database query generation through non-directed graphs supporting natural languages. The ontology language used in this system is based on the RDF structure. In order to construct queries, query terms are suggested to a user in a natural language from a predefined vocabulary. In a report of the EU Translational Research and Patient Safety in Europe (TRANSFoRm) [33], a query and data extraction workbench has been presented. The TRANSFoRm query formulation workbench software tool provides interfaces to *author*, *store* and *deploy* queries of clinical data in order to identify subjects for clinical studies. Moreover, TRANSFoRm query formulation workbench enables users to define criteria groups flexibly, whilst catering for complex queries with combinations of operators.

Table 2
Comparison of ontology-based query formulation tools/approaches.

Tools/approaches	Query support by semantic clause	Query support by text	Replication of data not required in ontology	Supports multimedia database	Heterogeneous data-sources support	Natural language query
CROEQS [34]	✔	✔	✔	✔		
GRQL [21]	✔	✔				
Ontogator [23]	✔	✔		✔		
OntoQF [25]	✔	✔	✔			
OntoViews [24]	✔	✔		✔		
Smartch [27]	✔	✔				
SEWASIE [22]	✔	✔	✔		✔	
TAMBIS [4,20]	✔	✔				
TRANSFoRm [33]	✔	✔	✔		✔	
VISAGE [26]	✔	✔	✔		✔	
Ontology and Natural Language [35]		✔	✔			✔
OPTIQUE [36]	✔	✔	✔		✔	
KIRA [37]	✔	✔	✔	✔	✔	
ATHENA [38]		✔	✔			✔
Using ontology SPARQL [39]	✔					
Pay-As-You-Go Method [40]	✔				✔	
Ontop [41,42]	✔		✔		✔	
Querying via OWL 2 QL [43]	✔		✔			

The search method of ontology-based image retrieval systems such as Ontogator [23], and OntoViews [24] are examples of a *concept-based multi-facet* search using RDFS ontologies. In a *multi-facet* search, multiple distinct views are augmented to data created via ontology projection [24]. OntoViews, supports semantic auto-completion of a query [24]. It uses a keyword search mechanism for ontology navigation. The search keywords are linked directly to ontology *classes*. A user search request is processed as a *multi-facet* search and results are delivered in a web browser. Once a single interesting *instance* (at least) has been found, additional information can be retrieved via ontology browsing.

Effective information retrieval is becoming more challenging with the increase in the use of Multimedia databases, which are usually bigger than traditional databases. In [34] a semantic search engine for multimedia databases namely CROEQS is presented that works as both ontology-based query translator and text based search engine. In relation to the use of ontologies for the provision of intelligent and accurate search engines Kunmei Wen [27] proposed Smartch, which is an ontology-based search engine. In this approach, a ranking method is proposed while searching for concepts, instances and the relationships between them. In Smartch, the end-users' search is performed by keywords. Once results are retrieved end users' can use the graphical user interface of Smartch to view all instances of an ontology concept, view relationship between two entities and view all instances of a user defined concept. Table 2 presents a comparison between some of the major ontology-based query formulation tools and approaches.

3.3. Ontology-based information linking approaches

The work carried out in the European TONES project [44] provides relational database access through ontologies. In this approach, data access is enabled by defining links between ontology *concepts* and *relational data*. This ontology-to-database mapping mechanism enables a designer to link a data source to an OWL-Lite ontology. While defining mappings, the designer needs to take into account that an ad-hoc identifier should denote each concept *instance* so that *instance values* cannot be confused with data items in the data source. Queries are formulated by consulting ontology-to-database mapping rules, but this rule derivation process is carried out manually by ontology and database experts

[44]. Another ontology-based information linking approach with similar techniques, but for query refinement purposes, is presented in [45,46]. This approach stores concepts from a data source as part of the ontology and links actual data with ontology concepts. The query answers are improved by using the semantic knowledge expressed in an ontology. Database queries are transformed by using *is-a*, *part-of* and *sync-of* relationships between ontology concepts.

The work carried out in designing ontology-based interactive information retrieval interfaces [47] provides an ontology-based web information retrieval system. This approach works as an interactive information retrieval system where end-users are guided through an ontology (OWL-based) driven graphical interface to define the search criteria. This work mainly addresses the problem of "where to start in the usage of an ontology-based IR interface"; that is, which elements of the ontology should be provided to the user to begin the search specification [47]. Accordingly, a user first selects a relevant domain in order to start building a query. The interface then provides a number of search entry points along with their descriptions. Once the user selects the desired ontology elements, web information elements are retrieved by following the static ontology-to-web links.

In the SemanticLIFE project [48], a front-end approach guides the users in generating data requests. The SemanticLIFE system integrates multiple data sources and stores them in an ontological repository. The Virtual Query component of the SemanticLIFE system allows semantic query writing on the ontological RDF-based repository. Users are provided with an overview about the system data through a *Virtual Data* component which stores the extracted metadata of the data sources in the form of an ontology. The approach provides a query engine, which recommends the query patterns according to the users' querying context. Since it is based on a common ontology mapped from the local data source ontologies, this approach can refine users' queries and create sub-queries over the local data sources.

3.4. Ontology-based query refinement approaches

Ontology-based query refinement approaches aim at enabling end-users to make an improved formulated query. These approaches attempt to improve information retrieval by replacing or adding extra terms into an initial query. Most of the existing

query refinement approaches include both query rewriting and expansion operations. Using these approaches end-users are provided interaction with candidate expansion terms based on concept hierarchies which stem naturally from the developed domain ontologies and associated ontological schema. This section discusses these ontology-based query refinement techniques that have been introduced over the past few years such as *Thesaurus Ontology Navigation* [49,50], *Ambiguity-Driven* [51,52], *Information-Need Driven* [53] etc.

Query expansion implementations (e.g. [49,50]) use *thesaurus ontology navigation* for query expansion. These approaches use the *WordNet* ontology (http://wordnet.priceton.edu) for query expansion and adapt basic keyword search mechanisms using keywords, which are identified in the ontology for a matching concept. Another approach based on this thesaurus ontology navigation approach is the Knowledge Sifter [29]. Knowledge Sifter is a scaleable agent-based system that supports access to heterogeneous information sources and relies on the agents technology for query refinement. In this approach, a user query formulation agent supports user query specification to access multiple ontologies using an integrated conceptual model expressed in the OWL. This user query formulation agent also consults the ontology agent to refine or to generalise a query based on the semantics provided by the ontology.

In QuOnto [54] and MASTRO [55], the query answering process is performed through query rewriting. Both of the MASTRO and QuOnto systems adapt a similar ontology based query answering service [55]. In these approaches, end-user queries are first reformulated on the basis of ontological intensional knowledge, and then they are evaluated by a database engine using a means of predefined mappings. Database views are defined for ontology concepts and roles using SQL queries are specified in ontology-to-database mapping declarations. In [56] an ontology-based tool to convert a natural language query into nRQL query has been proposed. To achieve the conversion, first a pre-populated dictionary is used to search the synonyms of query terms. If no matching records are found then the ontology search is performed, which results in extracting a sequence of entities represented in form of triples. Finally, nRQL query is generated based on the resultant information. The ontology-based query refinement approaches such as the *Step-By-Step Query Refinement* [52], examine query ambiguity in relation to both structural and semantic ambiguities. Structural ambiguity deals with the actual structure of a query that is analysed with respect to the underlying ontological knowledge. In the case where a conflict is detected, alternative suggestions are retrieved and presented to the end-user for selection.

4. Discussion: ontology-based information retrieval

In Section 2, benefits, weaknesses, power and effectiveness of current mature ontology development languages in relation to query formulation have been highlighted, and it has been noted that OWL has greater support for expressing semantics when compared to RDF and RDFS. According to the literature review of ontology-based information retrieval in Section 3, it appeared that the focus has been on (1) visual or interactive query formulation; (2) information linking approaches; and (3) query refinement approaches. Ontology based visual or interactive database query formulation systems use visual representations to express the search criteria. Most of these systems are based on the RDF structure and support the specialisation or generalisation of the base or filler ontology concepts in order to build database specific queries interactively. However, it may be concluded that much of this work has been (e.g., [57,21,29] etc.) towards interactive query generation through *nondirected* graphs. Other approaches (e.g.,

[44,46,58,48] etc.) store all data from a data-source as part of the ontology (as ontology instances) or link it directly to ontology concepts. But, often it is not practically feasible to store all data as part of a certain domain ontology, especially for systems with large amounts of data. Data that are stored as part of the ontology often need to be loaded in memory to perform the *Select* query operations. Moreover, this may become both a complex and time consuming activity to directly link all database instances to associated ontological concepts. Furthermore, it appears that a limited number of query formulation approaches reviewed in the literature review build on the assertion capabilities of ontologies. Thus, it can clearly be deduced that these approaches need to be extended to include further details, such as what needs to be included in the ontology from database along with the domain knowledge needed to initiate the query formulation process, to enable ontology-based query formulation based on the ontology semantic and assertion capabilities.

As discussed before, a domain ontology could be used for representing domain metadata with related semantics extracted from the relational database schema. To achieve this, first there is the need:

1. to identify the extent to which domain metadata and relationships from a relational database can be transformed into a domain ontology schema, and
2. to identify a systemic approach to transform this selected domain metadata and relationships into the domain ontology schema.

Moreover, a description logic based knowledge representation formalism is well suited for modelling domain knowledge (also called assertional knowledge) in a domain ontology in terms of concepts and properties. However it may not be necessary or even possible to use all of OWLs description logic constructs for formulating queries. Therefore, there is a need to identify OWL constructs that can be utilised to specify domain specific knowledge as concept restrictions for the purpose of formulating relational database queries. In addition, there are significant differences between OWL ontology statement constructs and relational query statement constructs. Furthermore, the OWL concept restrictions could be either simple or complex potentially involving many conditions. As a consequence, ontology-driven relational query formulation is not only dependent on the underlying relational database schema structure, but also on the translation of individual and different combinations of ontology statement constructs into relational query statement constructs.

An ontology guided (relational) query formulation process also needs to take into consideration aspects of ontology modelling, processing and integration of domain knowledge based on the underlying relational database models and mapping to ontological queries to a relational database schema. The following sections first review the state of the art in this field and build a case to inform the usability of combining both ontology-to-database transformation and mapping approaches for relational database query formulation.

5. Approaches to perform database-to-ontology transformations and to define ontology-to-database mappings

In order to specify the schema structure of a domain ontology for relational query formulation, one requirement is to represent the domain metadata along with the semantic relationships in the underlying relational database schema into the ontology schema. Thus, to represent a relational data model in an ontology model, the transformation of the relational model into the ontol-

ogy model remains an essential requirement in order to achieve correct transformations. This is because an ontology generally contains the definition of the concepts and their relationships for a given domain, as well as the domain rules (e.g. cardinality, disjointness etc.) that restrict the semantics of concepts and the conceptual relationships in a specific conceptualisation of a particular application domain. In contrast a relational data model, on the contrary, represents the structure and semantic data integrity of a given database application [59]. To provide detail this section reviews the existing database-to-ontology transformation and mapping approaches. These approaches are reviewed in terms of the loss of data and semantics, structural mapping, domain applicability and correctness.

Currently, there are several tools and approaches available that can be used to define mappings between ontology schema and database schema (called ontology-to-database mappings) such as D2R-MAP [60], extended D2R [61], R2O [62], VisAVis [63], in [64] and many others. These approaches are based on the assumption that both the database and the ontology pre-exist, and produce a set of corresponding mappings between the relational database schema and the ontology schema. These mapping approaches are different from the *transformation* approaches, which aim at generating an ontology model from a relational model (called database-to-ontology transformations) as described in [65–67]. A majority of these approaches provide trivial transformations, where each database *table* maps to an ontology *class*, each *column* to a *datatype* property, each *row* to an *instance* and *foreign key* columns are used to link an *instance* of a *class* to *instances* of another *class*. In addition to these approaches, several database-to-ontology transformation tools have been developed. For example, DataGenie [68] is a plug-in for Protégé [68] that imports data from a relational database to an ontology, D2RQ [69] treats Non-RDF relational databases as virtual RDF graphs, D2RMAP [70] is a database to RDF mapping language and processor, RDB2Onto [71] works by creating the semantic metadata from a relational database, RDB2ONT describes a formal algorithm to use the relational database metadata plus structural constraints to construct an OWL ontology, and in [72] an approach to develop ontologies from relational databases using reverse engineering is presented. The following sections discuss these approaches in more detail.

5.1. Ontology-to-database mapping approaches

The ontology-to-database mapping approaches assume the existence of both a relational database and an ontology and produce a set of corresponding mappings between them [73]. The related approaches are D2R-MAP [60], extended D2R [61], R2O [62], VisAVis [63] and in [64]. These mapping approaches relate each construct in the relational database schema to a construct in the ontology schema and ignore unrelated constructs between them. The R2O (Relational to Ontology) [62] approach, for example, is an extensible and declarative language to describe mappings between relational database schemas and ontologies implemented in RDF(S) or OWL. The most important aspect of this approach is the use of the database schema and the ontology without adaptation. Also, this approach defines a declarative specification of the mappings between its modelling components. The ontology-to-database mappings are defined as a set of mapping elements that relate a relational DB schema to an ontology schema. This means that database *tables*, *columns*, *primary* and *foreign keys*, etc., are related to domain ontology *concepts*, *attributes*, *relationships*, etc.

R2O is an extension of recent approaches such as D2R-MAP [60] and extended D2R [61], which lack expressiveness in terms of writing complex mapping statements and are not considered fully declarative [62]. As a consequence, the R2O mapping language has been considered sufficiently expressive to cope with complex mapping situations arising from low similarity between ontology and database models [62]. The mapping definitions generated by R2O are not intended to be generated manually, and therefore they cannot be read or updated without using its specific GUI. Such an ontology-to-database mapping approach is not sufficient to be used for ontology creation for query formulation.

An approach for automatic database to ontology mapping that satisfies both information and query preservations properties of semantic mapping is presented in [74]. Here, information preservation is the ability to recreate original database from mapping results and query preservation is the ability to translate each relational query over a relational database into an equivalent semantic query over resulting RDF graph. D2RQ [69] extracts the contents of a relational database to an RDF graph as per the mappings specified in a mapping language which is also expressed in RDF. Another tool that is heavily influenced by D2RQ [69] is Automapper [75]. Automapper creates an OWL ontology through SPARQL [76] to describe a relational schema. The feature that separates Automapper from other ontology mapping approaches is that the generated OWL ontology is also enhanced with SWRL [16] rules to express the constraints, such as the primary key or attribute datatype restrictions.

In [77] a three phased approach to extract ontology from a relational database is presented. In the first phase, ontology TBox is written using relational schema by generating classes from both referenced and referencing columns. Here, the referencing column is defined as a subclass of the one related to the referenced column. In the second phase, ABox is written using DB values, and in the final phase, reasoning is performed to extend the ontology.

In [78] an automated ontology construction approach is presented that considers relational schema. In addition to trivial table to class transformations, the proposed mappings in this approach include some additional rules in relation to a table key column(s) and other constraints. These include map foreign key column to ontology object property and non-foreign key column to ontology datatype property. Moreover, it maps *primary key* and *unique* constraint to ontology *InverseFunctionalProperty*, and NULL constraint to *minCardinality* constraint with *minCardinalty* of 1 [78]. Thus, this approach can be useful for the transformation of both relational database structure and its constraints to ontology. Table 3 presents a comparison between some of the major database to ontology mapping tools and approaches discussed in this section.

5.2. Database-to-ontology transformation approaches

The database-to-ontology transformation approaches assume that only a relational database exists while an ontology is produced from the relational database by applying certain transformation rules [73]. Such approaches include learning ontologies from relational databases [65], mappings from relational to OWL model [66], relational databases to OWL ontology [67] and rule-based transformation of SQL relational databases to OWL ontologies [84]. Most of these approaches result in specifying an ontology that has the same flat structure (i.e. *classes* and *instances*) as the original relational database (i.e. *relations* and *columns*). These approaches utilise automatic or semi-automatic transformations of relational databases to ontologies. Here the transformation is based on a set of rules specifying how to transform the constructs of a relational model to develop an ontology with the relational model as its domain. The basic transformations adopted in these approaches are very similar, where the constructs of a relational model such as *tables*, *columns*, *datatypes* and *constraints* are transformed in an ontology model as *classes*, *instances*, *properties* and *constraints*, respectfully. In such a scheme, a *table* is represented as a *class* unless all its *columns* are *foreign keys* to other *tables*. The reference *column* of a *foreign key* is represented as an *object* property. A col-

Table 3
Comparison of database to ontology mapping tools/techniques.

DB to ontology mapping tools/approaches	Ontology to DB mapping support	Mappings defined using ontology	Mappings defined using XML	Maps DB instances to ontology
D2R-MAP [60]	✔		✔	✔
R2O [62]	✔		✔	✔
VisAVis [63]	✔	✔		
OntoQF [30]	✔	✔		✔
Extended D2R [70]	✔		✔	✔
D2OMapper [68]	✔		✔	✔
Mapping DB to RDF and OWL [74]	✔	✔		✔
RDB2OWL [79]	✔	✔		
BOOTOX [80]	✔	✔		
RDF Graph from using SPARQL [81]	✔	✔		✔
RBA [82]	✔	✔		
R^2BA [83]	✔	✔		

umn is represented as a *datatype* property accompanied by a *maximum* cardinality of *one*, unless it is a *foreign key*. The column constraint *Unique* is represented as an *inverse* functional property. Similarly, *Not Null* is represented as a *minimum* cardinality of *one*. A *primary key* is represented as both an *inverse* functional property and a *minimum* cardinality of *one*. Finally, a *row* is represented as an ontology *instance*.

Another database-to-ontology transformation tool, RDB2ONT [85] can be used for generating OWL ontologies from relational database systems. RDB2ONT describes a formal algorithm that uses relational database metadata and structural constraints to construct an OWL ontology whilst preserving the structural constraints of the underlying database [85]. The RDB2ONT tool has two major components: the OWL Builder and the OWL Writer. The OWL Builder extracts metadata and structural constraints from a relational database system and builds a model. This model is then used to generate an OWL ontology describing the underlying database. This approach is less complex when compared to D2R-MAP [60], D2R [69] or R2O [22] which are based on creating a new complicated mapping language. This is mainly due to its simplicity and easy of configuration for defining concrete data mappings.

Triplify [86] is an RDF extraction tool from relational schemas, which uses SQL queries to select subsets of the database schema and maps them to ontology classes and properties. The advantages of the Triplify tool include predefined configurations for the schemas and the use of easily understandable SQL queries for the mapping representation. Another such tool is SquirrelRDF [87], which uses predefined mappings to extract database contents as RDF triples. Its use is fairly simple, but it does not offer the wide range of features when compared to D2RQ [69] as explained earlier. Later, Tether [88] has provided some refinements and solved several shortcomings of the native translation of a relational database to RDF approach. Most of these refinements; such as to reduce the size of the RDF graph, were specific to cultural heritage domain.

In OntoQF [25] an ontology is generated from a relational database (model) that stores the domain metadata, semantics and related domain knowledge. The major differences of this approach with other approaches is that in OntoQF (1) the transactional data remains at the original data source(s); (2) in can handle the database relations up to third normal form; and (3) the resultant transformation describes the original database relationships [25]. Moreover, in OntoQF, the ontological descriptions are based on domain metadata objects; and consequently, it supports changes and extensions to the underlying database schema. In [89], modelling of health service knowledge is done in terms of health service ontology, where the health service ontology is defined from the perspective of concept hierarchy and ontology concepts. Moreover, the matchmaking algorithm in this approach is powered by UMLS ontology. In [90] graph-based knowledge representation model has been used in query answering; in particular, a conceptual graph model is used instead of knowledge representation formalisms as used in [25].

Table 4 presents a feature comparison between some of the major database to ontology transformation tools and approaches discussed in this section.

5.3. Ontology-to-conceptual data model

Ontologies allow an interaction between data held in different formats and can potentially be used as the basis to guide and validate models of particular domains. For example, a considerable amount of work has been reported which aims to transform ontologies to conceptual data models (expressed, for example, in UML or in ER) in [98,99,72]. This section reviews the key relevant and mature ontology-to-database mapping approaches.

According to El-Ghalayini et. al. [98], a domain ontology can be mapped to a domain conceptual data model (CDM). In this research, several mapping rules have been proposed that guide

Table 4
Comparison of database to ontology transformation tools/techniques.

Database to ontology transformation tools and/or approaches	DB to ontology transformation support	Output ontology in OWL format	Ontology extension by reasoning	Preserve structural constraints
RDB2Onto [71]	✔	✔		
RDB2ONT [85]	✔	✔		✔
Learning ontology from relational schema [78]	✔	✔		✔
Generating OWL ontology from RDB [77]	✔	✔	✔	✔
Learning ontology from RDB [65]	✔	✔	✔	✔
DB2OWL [91]	✔	✔		✔
OntoQF [25]	✔	✔	✔	✔
Mapping RDB to OWL structure [92]	✔	✔		✔
Transforming of RDB to OWL [93]	✔	✔		✔
RDB2RDF [94]	✔			✔
Mapping RDB into Ontology [95,96]	✔	✔		✔
A systematic mapping via reverse engineering [97]	✔	✔		✔

the transformation from a given domain ontology to a corresponding conceptual schema. Another approach to building ontologies from relational databases using reverse engineering is presented in [72]. Unlike RDB2ONT [85] and RDB2Onto [71], this approach does not directly transform a relational database into an ontology; rather it uses an *entity relationship model* to reverse engineer a domain ontology. In this approach, OWL-DL is used as an ontology representation language and ER for data modelling. The graph transformation language, as in [100], is used for *node* and *edge* addition into an ER model. The *node* addition operation is used to introduce new *objects* into the ER model and the *edge* addition operation to build *relationships* between ER objects.

Recently, an approach to transform a domain ontology into a relational database has been presented [99] based on an algorithm embedded in the OWL2DB [101]. In this approach, OWL documents are parsed in order to generate the corresponding Data Definition Language (DDL) scripts. During the parsing and data transformation process, the system first transforms ontology classes into database table definitions; the next steps are transformations of object, datatype properties and constraints into complete DDL statements, and finally, the database is filled with class instances. In order to transform ontology *classes* into relational database *tables*, the approach uses a *breadth-first* search on the hierarchical levels of the ontology classes [101]. As a result, one *table* in a relational database is created for each *class* in an ontology with a *one-to-one* relationship between classes and their subclasses. The OWL *object* properties are transformed into table *relationships*, which again uses *breadth-first* search. Depending upon the local cardinality of *class* properties, *one-to-many* or *many-to-many* relationships between tables are created. This approach can transform all OWL-Lite syntax but only part of OWL-DL syntax.

6. Discussion: database-to-ontology transformation and mappings approaches for relational query formulation

An ontology generally contains the definition of the concepts and their relationships for a given domain, as well as the assertions and domain rules (e.g. *cardinality*, *disjointness* etc.) that restrict the semantics of concepts and the conceptual relationships in a specific conceptualisation of a particular application domain. A relational data model, on the contrary, represents the structure and integrity of the data elements of the, in principle, *single* specific enterprise application(s) by which it is being used. A relational database is a collection of *relations* (*tables*) and a relation consists of a relational schema and a relation instance. The relation schema in a database consists of the schemas for the relations, whereas the relation instances are sets of tuples, also called records. Moreover, each tuple is a row and all rows have the same number of fields (columns). After reviewing such differences and similarities between ontological and relational data models it has been noted that ontologies are semantically richer than database (conceptual) schemas, because conceptual data models only aim at establishing a link between users and domain requirements, and describe a logical structure of the data. An ontology can also be used to specify domain knowledge of a specific domain of interest. Moreover, ontologies can play a significant role in information system development and have the ability to represent conceptual data models using ontological theories, for example as reported by [98,30]. Therefore, existing relational data models can be used to create ontologies, while existing ontologies can be used to generate relational (conceptual) schemas.

Ontology-to-database mapping approaches assume the existence of both a relational database and a domain ontology. However, the database-to-ontology transformation approaches assume that only a relational database exists and an ontology is generated by applying database-to-ontology transformation rules. According to the database-to-ontology transformation approaches as discussed in this paper, the process of ontology construction from relational databases involves analysing the database schemas to determine the database-to-ontology transformation dependencies. This analysis helps in determining the relational entities that may be transformed into ontology concepts. It also helps to group together or separate in occasions the information specified in a relational database table and to determine relationships between different tables. However, the database-to-ontology transformation rules are solely dependent on application requirements. Most of the existing database-to-ontology transformation approaches do not provide an exact representation of the domain-metadata in an ontology and also do not enable the generation of the respective database relations. Therefore, such transformation approaches do not particularly aid in the process of specifying concept restrictions, and also when generating complex database queries. In order to overcome these shortcomings and to use ontology definitions for database information retrieval, a combination of both database-to-ontology transformation and ontology-to-database mappings [25] may be utilised, which could be an interesting future challenge for the research community. More areas of future research are outlined in the following section.

7. Future opportunities and challenges

Most of the existing approaches are based on using single domain ontology for generating relational database queries. Using such approach, database domain metadata and semantics can be transformed into a domain ontology schema with domain knowledge added as concept restrictions. It is possible to adapt the same approach for multiple databases that are geographically distributed, but conceptually related with respect to a common subject area; for example, *a database that manages patients treatment records and another that manages patients family history, where the domain-subject is Patients*. In such cases, an integrated ontology needs to be developed to capture the common as well as the different domain metadata and related semantics of the underlying distributed databases. Moreover, the existing query formulation frameworks should be extended to develop an approach to integrate related domain knowledge in order to obtain a *unified* domain ontology that captures all of the domain concepts.

Moreover, in the past the focus has been on visual or interactive query formulation, information linking and query refinement approaches. Most of these ontology based visual or interactive database query formulation systems either use visual representations to express the search criteria or they are based on the ontology structure and support the specialisation or generalisation of ontology concepts in order to build database specific queries interactively. Other approaches store all data from a data-source as part of the ontology or link it to ontology concepts. However, it appeared that, with the exception of few approaches, there is still a lack of knowledge-driven query formulation approaches that build on the assertion capabilities of OWL-DL such as OWL-2. Moreover, there is a need to extend the existence approaches to answer the questions like: *What needs to be included in the ontology from database along with the domain knowledge needed to initiate the query formulation process, to enable ontology-based query formulation based on the OWL-DL semantic and assertion capabilities.* Plus *how such domain knowledge can be automatically evolved and extended within the existing domain ontology?* Furthermore, this work may be extended to further domains; and in particular, for enhancing the searching capabilities of massively loaded information management systems such as *national statistical survey portals* and *context-aware environments* that mobile devices are part of.

As per the literature review presented in this paper, it also appeared that currently there is very limited tool support available for the direct specification and manipulation of ontology domain knowledge for query formulation. In most of the presented systems; domain experts, with the help of knowledge engineers express domain knowledge in terms of ontology statements that include the definition of property constraints for concepts and individuals. In this way, whenever related knowledge about real world concepts change or when new knowledge is added into the domain ontology, the related ontology file is reloaded into the ontology server. In order to enable domain experts to directly specify and manipulate domain knowledge, it would be an interesting future work to empower the ontology knowledge component of such systems with tool support for the direct *(on the fly)* specification and manipulation of related domain knowledge in the ontology server. However, leaving ontology knowledge specifications completely for domain experts, who usually do not have description logic experience, may end-up in defining inconsistent domain knowledge specifications from real world knowledge, and hence this needs to be controlled.

There are also various challenges associated with the management of large data sets (so-called Big data) include structuring, search, analysis, visualisation e.g. as detailed in [102,103]. Due to the large and complex data sets it becomes difficult to process stored information using traditional data management tools and processing applications. Currently there is little research done on investigating efficient ways to process these large data sets to get benefits such as: search performance, creation of reference data and enable reasoning. One of the best possible ways to achieve this is by building ontological knowledge base for Big data; and that is to (a) define a semantic model of data, (b) specify domain knowledge, and (c) define links between different types of semantic knowledge. Hence, ontologies can be used to discover information from Big Data. Based on these clear prospects, future research efforts can be towards an investigation of the feasibility to use ontological knowledge base for the specification of Big datas meta data as a foundation to efficiently discover useful information for analysis. Such research efforts will need to find answers for research questions such as: (1) *To what extent the meta data from Big data (NoSQL DB) can be extracted and aggregated?* and (2) *How can the extracted Big datas meta data along with domain knowledge be represented to be the foundation of knowledge discovery?*

References

[1] G. Zhang, T. Siegler, P. Saxman, N. Sandberg, R. Mueller, N. Johnson, D. Hunscher, S. Arabandi, Visage: a query interface for clinical research, Summit Transl. Bioinform. 1 (2) (2010) 3.

[2] D. Damljanovic, M. Agatonovic, H. Cunningham, Freya: an interactive way of querying linked data using natural language, in: Extended Semantic Web Conference, Springer, Berlin, Heidelberg, 2011, pp. 125–138.

[3] J. Fan, G. Li, L. Zhou, Interactive sql query suggestion: making databases user-friendly, in: IEEE 27th International Conference on In Data Engineering (ICDE), IEEE, 2011, pp. 351–362.

[4] N. Paton, R. Stevens, P. Baker, C. Goble, S. Bechhofer, A. Brass, Query processing in the tambis bioinformatics source integration system, in: Proceedings of the IEEE International Conference on Scientific and Statistical Databases (SSDBM), 1999, pp. 138–147.

[5] A. Meštrović, A. Calì, An ontology-based approach to information retrieval, in: Semantic Keyword-based Search on Structured Data Sources, Springer, 2016, pp. 150–156.

[6] F. Ramli, S. Noah, T. Kurniawan, Ontology-based information retrieval for historical documents, in: Third International Conference on Information Retrieval and Knowledge Management (CAMP), IEEE, 2016, pp. 55–59.

[7] T.R. Gruber, A translation approach to portable ontology specifications, Knowl. Acquis. 5 (1993) 199–220, http://dx.doi.org/10.1006/knac.1993.1008.

[8] K. Munir, K. Hasham, R. McClatchey, Development of a large-scale neuroimages and clinical variables data atlas in the neugrid4you (n4u) project, J. Biomed. Inform. 57 (2015) 245–262.

[9] Harold, E. Rusty, Effective XML, Addison-Wesley, 2004.

[10] K. Munir, M. Waseem Hassan, A. Ali, R. McClatchey, I. Willers, Database independent migration of objects into an object-relational database, in: The

[11] G. Klyne, J. Carroll, Resource description framework (rdf): concepts and abstract syntax, Ph.D. thesis, 2004.

[12] A. Gómez-Pérez, O. Corcho, Ontology languages for the semantic web, IEEE Intell. Syst. 17 (1) (2002) 54–60.

[13] P. Hayes, I. Horrocks, F. Harmelen, OWL web ontology language; semantics and abstract syntax, W3C, 2002.

[14] W3C, OWL 2 web ontology language, world wide web consortium (W3C), 2017.

[15] Y. Gil, V. Ratnakar, A comparison of (semantic) markup languages, in: Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference, 2002, pp. 413–418.

[16] I. Horrocks, P. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, M. Dean, SWRL: a semantic web rule language swrl: a semantic web rule language combining OWL and RuleML, W3C Member Submission, 2004.

[17] D. Embley, Nfql: the natural forms query language, in: ACM Transactions on Database Systems, vol. 14, 1989, pp. 168–211.

[18] R. Semmel, An integrated system for query formulation and database design, in: Proceedings of 4th International Conference on Software Engineering and Knowledge Engineering, IEEE, 1992, pp. 40–46.

[19] M. Scamell, A human factors experimental comparison of sql and qbe, in: IEEE Transactions on Software Engineering, vol. 19, 1993, pp. 390–402.

[20] P. Baker, A. Brass, S. Bechhofer, C. Goble, N. Paton, R. Stevens, TAMBIS: transparent access to multiple bioinformatics information sources, in: Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology, 1998, pp. 25–34.

[21] N. Athanasis, V. Christophides, D. Kotzinos, Generating on the fly queries for the semantic web: the ics-forth graphical RQL interface (GRQL), in: Proceedings of the 3rd International Semantic Web Conference, 2004, pp. 486–501.

[22] T. Catarci, T. Di Mascio, E. Franconi, G. Santucci, S. Tessaris, An ontology based visual tool for query formulation support, in: On The Move to Meaningful Internet Systems 2003: OTM 2003 Workshops, Springer, Berlin/ Heidelberg, 2003, pp. 32–33.

[23] E. Hyvonen, S. Saarela, K. Viljanen, Ontogator: combining view and ontology-based search with semantic browsing, in: Proceedings of XML Finland, 2003.

[24] E. Makela, E. Hyvonen, S. Saarela, K. Viljanen, Ontoviews – a tool for creating semantic web portals, in: Semantic Computing Research Group Helsinki Institute for Information Technology (HIIT), 2003.

[25] K. Munir, M. Odeh, R. McClatchey, Ontology-driven relational query formulation using the semantic and assertional capabilities of owl-dl, Knowl.-Based Syst. 35 (0) (2012) 144–159, http://dx.doi.org/10.1016/j.knosys.2012.04.020.

[26] G. Zhang, T. Siegler, P. Saxman, N. Sandberg, R. Mueller, N. Johnson, D. Hunscher, S. Arabandi, Visage: a query interface for clinical research, AMIA Clin. Res. Inform. Summit, San Francisco (2010) 76–80.

[27] K. Wen, R. Li, B. Li, Searching concepts and association relationships based on domain ontology, in: 9th International Conference on Grid and Cooperative Computing (GCC), 2010, pp. 432–437.

[28] K. Munir, M. Odeh, R. McClatchey, S. Khan, I. Habib, Semantic information retrieval from distributed heterogeneous data sources, in: The 4th International Workshop on Frontiers of Information Technology, Special Track on Bioinformatics for Academia and Industry, 2006.

[29] L. Kerschberg, M. Chowdhury, A. Damiano, Knowledge sifter: ontology-driven search over heterogeneous databases, in: Proc. 16th Int. Conf. Scientific and Statistical DB Management, 2004.

[30] K. Munir, M. Odeh, R. McClatchey, Managing the mappings between domain ontologies and database schemas when formulating relational queries, in: Proceedings of the 2009 International Database Engineering and Applications Symposium, IDEAS '09, ACM, New York, NY, USA, 2009, pp. 131–141, doi: http://doi.acm.org/10.1145/1620432.1620446..

[31] K. Munir, M. Odeh, P. Bloodsworth, R. McClatchey, Using assertion capabilities of an owl-based ontology for query formulation, in: IEEE, International Conference on Information and Communication Technologies (ICTTA08): From Theory to Applications, IEEE, 2008, pp. 1–6, http://dx.doi.org/10.1109/ICTTA.2008.4530296.

[32] E. Kapetanios, P. Groenewoud, Query construction through meaningful suggestions of terms, in: FQAS, 2002, pp. 226–239.

[33] L. Zhao, S. Lim Choi Keung, J. Rossiter, T. Arvanitis, Report for the eu translational research and patient safety in europe (transform) project: query formulation workbench, 2012.

[34] S. Vandamme, J. Deleu, T. Wauters, B. Vermeulen, F. De Turck, Croeqs: Contemporaneous role ontology-based expanded query search – analysis of the result set size, 10th Workshop on Image Analysis for Multimedia Interactive Services, 2009 (WIAMIS '09), 2009, pp. 169–172.

[35] B. Sujatha, S.V. Raju, Ontology based natural language interface for relational databases, Proc. Comput. Sci. 92 (2016) 487–492.

[36] E. Kharlamov, S. Brandt, E. Jimenez-Ruiz, Y. Kotidis, S. Lamparter, T. Mailis, C. Neuenstadt, Ö. Öezçep, C. Pinkel, C. Svingos, et al., Ontology-based integration of streaming and static relational data with optique, in: Proceedings of the 2016 International Conference on Management of Data, ACM, 2016, pp. 2109–2112.

[37] F. Amato, V. Moscato, A. Picariello, G. Sperlí, Kira: a system for knowledge-based access to multimedia art collections, in: 2017 IEEE 11th International Conference on Semantic Computing (ICSC), IEEE, 2017, pp. 338–343.

2nd IEEE International Workshop on Autonomous Decentralized System, IEEE, 2002, pp. 132–139, http://dx.doi.org/10.1109/IWADS.2002.1194661.

[38] D. Saha, A. Floratou, K. Sankaranarayanan, U.F. Minhas, A.R. Mittal, F. Özcan, Athena: an ontology-driven system for natural language querying over relational data stores, Proc. VLDB Endowment 9 (12) (2016) 1209–1220.

[39] M.A. Hazber, R. Li, X. Gu, G. Xu, Y. Li, Semantic sparql query in a relational database based on ontology construction, in: 2015 11th International Conference on Semantics, Knowledge and Grids (SKG), IEEE, 2015, pp. 25–32.

[40] J.F. Sequeda, D.P. Miranker, A pay-as-you-go methodology for ontology-based data access, IEEE Internet Comput. 21 (2) (2017) 92–96.

[41] M. Rodriguez-Muro, R. Kontchakov, M. Zakharyaschev, Ontology-based data access: ontop of databases, in: International Semantic Web Conference, Springer, 2013, pp. 558–573.

[42] D. Calvanese, B. Cogrel, S. Komla-Ebri, R. Kontchakov, D. Lanti, M. Rezk, M. Rodriguez-Muro, G. Xiao, Ontop: answering sparql queries over relational databases, Semantic Web 8 (3) (2017) 471–487.

[43] S. Klarman, T. Meyer, Querying temporal databases via owl 2 ql, in: International Conference on Web Reasoning and Rule Systems, Springer, 2014, pp. 92–107.

[44] D. Calvanese, G. Giacomo, D. Lembo, M. Lenzerini, A. Poggi, R. Rosati, Linking data to ontologies: the description logic DL-LiteA, in: Proc. of the 2nd Workshop on OWL: Experiences and Directions (OWLED), 2006.

[45] C. Necib, J.-C. Freytag, Ontologies for database query reformulation, in: Advances in Databases and Information Systems (ADBIS), 2004.

[46] C. Necib, J.-C. Freytag, Query processing using ontologies, in: CAiSE, 2005, pp. 167–186.

[47] Garcia, M.-A.E. Sicilia, Designing ontology-based interactive information retrieval interfaces, Lecture Notes Comput. Sci. 2889 (2003) 152–165.

[48] H. Hoang, M. Nguyen, A. Tjoa, A. Andjomshoaa, A front-end approach for user query generation and information retrieval in the semanticlife framework, in: Proceedings of the 8th International Conference on Information Integration and Web-based Applications and Services, Austrian Computer Society, 2006, pp. 107–116.

[49] D. Buscaldi, P. Rosso, E. Arnal, A wordnet-based query expansion method for geographical information retrieval, in: Working Notes for the CLEF Workshop, 2005.

[50] M. Rila, The use of wordnet in information retrieval, in: ACL Workshop on the Usage of WordNet In Natural Language Processing Systems, 1998, pp. 31–37.

[51] N. Stojanovic, J. Gonzalez, L. Stojanovic, Ontologer – a system for usage-driven management of ontology-based information portals, in: Proc. L-CAP '03 Conference, 2003.

[52] N. Stojanovic, R. Studer, L. Stojanovic, An approach for step-by-step query refinement in the ontology-based information retrieval, in: Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence, WI '04, IEEE Computer Society, Washington, DC, USA, 2004, pp. 36–43.

[53] N. Stojanovic, Information-need driven query refinement, in: Proc. IEEE/WIC Int. Conf. Web Intelligence, 2003.

[54] A. Acciarri, D. Calvanese, G. Giacomo, D. Lembo, M. Lenzerini, M. Palmieri, R. Rosati, QUONTO: querying ontologies, in: Proc. of AAAI, 2005, pp. 1670–1673.

[55] A. Poggi, M. Ruzzi, Ontology-based data access with MASTRO, in: Proceedings of the 15th Italian Conf. on Database Systems (SEBD), 2007.

[56] H. Boumechaal, Z. Boufaida, Formalization of natural language queries, in: International Symposium on Innovations in Intelligent Systems and Applications (INISTA), 2011, pp. 495–499.

[57] E. Kapetanios, D. Baer, B. Glaus, P. Groenewoud, Data querying and analysis through integration of intentional and extensional semantics, in: 16th International Conference on Scientific and Statistical Database Management (SSDBM), 2004, pp. 353–356.

[58] A. Borgida, R.J. Brachman, Loading data into description reasoners, in: The ACM SIGMOD, International Conference on Management of Data, Washington, DC, USA, 1993, pp. 217–226.

[59] K. Munir, S.L. Kiani, K. Hasham, R. McClatchey, A. Branson, J. Shamdasani, Provision of an integrated data analysis platform for computational neuroscience experiments, J. Syst. Inform. Technol. 16 (3) (2014) 150–169, http://dx.doi.org/10.1108/JSIT-01-2014-0004. https://doi.org/10.1108/JSIT-01-2014-0004.

[60] C. Bizer, D2R MAP – a database to RDF mapping language, 2003.

[61] J. Barrasa, Ó. Corcho, A. Gómez-Pérez, A case study of database-to-ontology mapping, in: Semantic Integration Workshop (ISWC 2003), 2003.

[62] J. Barrasa, O. Corcho, G. Shen, A. Gomez-Perez, R2O, an extensible and semantically based database-to-ontology mapping language, in: 2nd Workshop on Semantic Web and Databases (SWDB), 2004.

[63] N. Konstantinou, D.-E. Spanos, M. Chalas, E. Solidakis, N. Mitrou, VisAVis: an approach to an intermediate layer between ontologies and relational database contents, in: International Workshop on Web Information Systems Modeling (WISM), 2006.

[64] Y. An, A. Borgida, J. Mylopoulos, Inferring complex semantic mappings between relational tables and ontologies from simple correspondences, in: Proceedings of On The Move to Meaningful Internet Systems (OTM'05): CoopIS, DOA, and ODBASE, Springer Verlag, 2005, pp. 1152–1169.

[65] M. Li, X. Du, S. Wang, Learning ontology from relational database, in: Proceedings of the 4th International Conference on Machine Learning and Cybernetics, 2005, pp. 3410–3415.

[66] G. Shen, Z. Huang, X. Zhu, X. Zhao, Research on the rules of mapping from relational model to OWL, in: Proceedings of the Workshop on OWL: Experiences and Directions, 2006.

[67] A. Buccella, M. Penabad, F. Rodriguez, A. Farina, A. Cechich, From relational databases to owl ontologies, in: Proceedings of the 6th National Russian Research Conference, 2004.

[68] Z. Xu, S. Zhang, Y. Dong, Mapping between relational database schema and owl ontology for deep annotation, in: Web Intelligence, 2006 (WI 2006), IEEE/WIC/ACM International Conference on Web Intelligence, 2006, pp. 248–552.

[69] C. Bizer, A. Seaborne, D2RQ – treating non-rdf databases as virtual rdf graphs, in: Proceedings of the 3rd International Semantic Web Conference (ISWC2004), 2004.

[70] C. Bizer, Database to RDF mapping language and processor, d2rmap, 2016. <http://www4.wiwiss.fu-berlin.de/bizer/d2rmap/d2rmap.htm>.

[71] M. Seleng, M. Laclavik, Z. Balogh, L. Hluchy, Rdb2onto: approach for creating semantic metadata from relational database data, in: Proceedings of the Ninth International Conference on Informatics, Slovak Society for Applied Cybernetics and Informatics, Bratislava, 2007, pp. 113–116.

[72] J. Trinkunas, O. Vasilecas, Building Ontologies from Relational Databases Using Reverse Engineering Methods, ACM, 2007.

[73] I. Astrova, N. Korda, A. Kalja, Rule-based transformation of sql relational databases to owl ontologies, in: Proceedings of the 2nd International Conference on Metadata & Semantics Research, 2007.

[74] J.F. Sequeda, M. Arenas, D.P. Miranker, On directly mapping relational databases to rdf and owl, in: Proceedings of the 21st International Conference on World Wide Web, WWW '12, ACM, New York, NY, USA, 2012, pp. 649–658.

[75] D.-M. Fisher, M., G. Joiner, Use of OWL and SWRL for semantic relational database translation, in: Proceedings of the Fourth OWLED Workshop on OWL: Experiences and Direction, 2008.

[76] A. Seaborne, Sparql query language for rdf, W3C Working Draft, 12 October 2004.

[77] J.W. Choi, M.H. Kim, Generating owl ontology from relational database, in: Third FTRA International Conference on Mobile, Ubiquitous, and Intelligent Computing (MUSIC), 2012, pp. 53–59.

[78] L. Yiqing, L. Lu, L. Chen, Automatic learning ontology from relational schema, in: IEEE Symposium on Robotics and Applications (ISRA), 2012, pp. 592–595.

[79] K. Čerāns, G. Būmans, Database to ontology mapping patterns in rdb2owl lite, in: International Baltic Conference on Databases and Information Systems, Springer, 2016, pp. 35–49.

[80] E. Jiménez-Ruiz, E. Kharlamov, D. Zheleznyakov, I. Horrocks, C. Pinkel, M.G. Skjæveland, E. Thorstensen, J. Mora, Bootox: practical mapping of rdbs to owl 2, in: International Semantic Web Conference, Springer, 2015, pp. 113–132.

[81] A. Oudani, M. Bahaj, I. Cherti, C. Luo, T. He, X. Zhang, Z. Zhou, Y. Ouyang, Y. Ling, Q. Liu, et al., Creating an rdf graph from a relational database using sparql, JSW 10 (4) (2015) 384–391.

[82] L.E.T. Neto, V.M.P. Vidal, M.A. Casanova, J.M. Monteiro, R2rml by assertion: a semi-automatic tool for generating customised r2rml mappings, in: Extended Semantic Web Conference, Springer, 2013, pp. 248–252.

[83] R. Berardi, V.M.P. Vidal, M.A. Casanova, R2ba-rationalizing r2rml mapping by assertion, in: ICEIS (2), Citeseer, 2015, pp. 5–14.

[84] I. Astrova, N. Korda, A. Kalja, Rule-based transformation of SQL relational databases to OWL ontologies, in: 2nd International Conference on Metadata Semantic Research, 2007.

[85] Q. Trinh, K. Barker, R. Alhajj, RDB2ONT: A tool for generating owl ontologies from relational database systems, in: Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT/ICIW 2006), 2006.

[86] S. Auer, S. Dietzold, J. Lehmann, S. Hellmann, D. Aumueller, Triplify: light-weight linked data publication from relational databases, in: Proceedings of the 18th International Conference on World Wide Web, ACM, 2009, pp. 621–630.

[87] S.D. Seaborne, A., S. Williams, SQL-RDF, in: W3C Workshop on RDF Access to Relational Databases, 2007.

[88] K. Byrne, Having triplets – holding cultural data as RDF, in: Proceedings of the ECDL 2008 Workshop on Information Access to Cultural Heritage, 2008.

[89] H. Dong, F. Hussain, Semantic service matchmaking for digital health ecosystems, Knowl.-Based Syst. 26 (6) (2011) 761–774.

[90] R. Thomopoulos, J.-R. Bourguet, B. Cuq, A. Ndiaye, Answering queries that may have results in the future: a case study in food science, Knowl.-Based Syst. 23 (5) (2010) 491–495.

[91] N. Cullot, R. Ghawi, K. Yétongnon, Db2owl: A tool for automatic database-to-ontology mapping, in: Proceedings of 15th Italian Symposium on Advanced Database Systems (SEBD 2007), 2007, pp. 491–494.

[92] N. Gherabi, K. Addakiri, M. Bahaj, Mapping relational database into owl structure with data semantic preservation, Comput. Sci. Inform. Security 10 (1) (2012).

[93] M. Dadjoo, E. Kheirkhah, An approach for transforming of relational databases to owl ontology, arXiv:1502.05844, 2015 Feb 20.

[94] P.T.T. Thuy, N.D. Thuan, Y. Han, K. Park, Y.-K. Lee, Rdb2rdf: completed transformation from relational database into rdf ontology, in: Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication, ACM, 2014, p. 88.

[95] M.A. Hazber, R. Li, Y. Zhang, G. Xu, An approach for mapping relational database into ontology, in: 2015 12th Web Information System and Application Conference (WISA), IEEE, 2015, pp. 120–125.

[96] M.A. Hazber, R. Li, X. Gu, G. Xu, Integration mapping rules: transforming relational database to semantic web ontology, Appl. Math. 10 (3) (2016) 1–21.

[97] A.A. Abbasi, N. Kulathuramaiyer, A systematic mapping study of database resources to ontology via reverse engineering, Asian J. Inform. Technol. 15 (4) (2016) 730–737.

[98] H. El-Ghalayini, M. Odeh, R. McClatchey, T. Solomonides, Reverse engineering domain ontologies to conceptual data models, in: Proceedings of the 23rd IASTED International Conference on Databases and Applications, 2005, pp. 222–227.

[99] E. Vysniauskas, L. Nemuraite, Transforming ontology representation from OWL to relational database, Inform. Technol. Control 35 (3A) (2006).

[100] P. Mitra, G. Wiederhold, M. Kersten, A graph oriented model for articulation of ontology interdependencies, in: Proc. Extending Database Technologies, Berlin Heidelberg, 2000, pp. 86–100.

[101] A. Gali, C. Chen, K. Claypool, R. Uceda-Sosa, From ontology to relational databases, Conceptual Model. Adv. Appl. Domains 3289 (2005) 278–289.

[102] M. Bilal, L.O. Oyedele, J. Qadir, K. Munir, S.O. Ajayi, O.O. Akinade, H.A. Owolabi, H.A. Alaka, M. Pasha, Big data in the construction industry: a review of present status, opportunities, and future trends, Adv. Eng. Inform. 30 (3) (2016) 500–521, http://dx.doi.org/10.1016/j.aei.2016.07.001.

[103] M. Bilal, L.O. Oyedele, J. Qadir, K. Munir, O.O. Akinade, S.O. Ajayi, H.A. Alaka, H.A. Owolabi, Analysis of critical features and evaluation of bim software: towards a plug-in for construction waste minimization using big data, Int. J.

Sustain. Build. Technol. Urban Develop. 6 (4) (2015) 211–228, http://dx.doi.org/10.1080/2093761X.2015.1116415.

Dr. Kamran Munir has BSc, M.Sc. and Ph.D. in Computer Science from United Kingdom. His primary research interests include Information Science, Big data, cloud computing, distributed data processing and knowledge management. He has contributed to the various European Commission projects: Asia Link STAFF (2004–2006), Health-e-Child (2006–2010), neuGRID (2011–2014) and neuGRID4You (2013–2016) in which he led the Joint Research Area. His role also includes the production and leadership of UK and international postgraduate computer science degree courses such as Big Data and Cloud Computing, and frequent collaborations with students and graduates, including the conduct of collaborative work and supervision of MPhil and PhD theses.

M. Sheraz Anjum has obtained MS degree from National University of Sciences and Technology (NUST) and currently doing PhD in Computer Science. His research interests are in the areas of knowledge management, relational databases and Big data.