International Journal of Scientific Engineering and Technology (ISSN : 2277-1581)
Volume No.2, Issue No.9, pp : 919-923 1 Sept. 2013

Ontology-Based Query Processing In a Dynamic Data Integration System

Md Mahmudur Rahman
Dept. of CSE, International Islamic University Chittagong, Chittagong, Bangladesh.
Email: mmr@cse.iiuc.ac.bd

Abstract

Data integration is concerned with unifying data that share some common semantics but originate from unrelated sources. Necessarily, when we work on data integration, we must take into account a more important and complex concept called "heterogeneity". We begin by introducing Data integration systems (DIS), Ontologies and other preliminary concepts that will be used throughout the presentation. We will discuss how can Ontologies support, Integration? Also about the Ontologies and integration problems. We define architecture used for Semantic Query Distribution and we formally discuss the problem of query rewriting inside our data integration framework. Finally, we close this paper illustrating the current problem future work.

Keywords

Data Integration, Query, Ontology, Mapping. etc

1. INTRODUCTION

Data Integration systems attempt to provide users with seamless and flexible access to information from multiple autonomous, distributed and heterogeneous data sources through a unified query interface. Ideally, a data integration system should allow users to specify *what* information is needed without having to provide detailed instructions on *how* or from where to obtain the information. Data integration is the problem of combining data residing at different sources, and providing the user with a unified view of these data. The problem of designing data integration systems is important in current real world applications, and is characterized by a number of issues that are interesting from a theoretical point of view. This paper is focused on some of these theoretical issues, with special emphasis on the following topics.

Data Integration is mainly composed by the global schema, the data sources and the mapping between them.Data sources store the information of the system and their structure is described by a source schema. The global schema is a virtual integration of these source schemas, in order to provide a unified view to the user. It works like a mediator between the user and the sources, allowing users to query the mediator and receive results from data sources.

2. Data Integration

A data integration system I is a triple (G,S,M) where:

• *G* is the global schema expressed in a language LG over an alphabet *A*G. The alphabet comprises a symbol for each element of G (i.e. a relation if G is relational, a concept or role if G is a Description Logic, etc.)

• S is the source schema, expressed in a language *LS* over an alphabet *AS*. The alphabet *AS* includes a symbol for each element of the sources.

• M is the mapping between *G* and *S* constituted by a set of assertions in the forms:

$$qS \rightsquigarrow qG$$
$$qG \rightsquigarrow qS$$

where *qS* and *qG* are two queries of the same arity, respectively over the source schema S and over the global schema G. Queries *qS* are expressed in a query language $L_{M,S}$ over an alphabet A_S,

Queries *qG* are expressed in a query language L_G over an alphabet A_G. Intuitively, an assertion

$qS \rightsquigarrow qG$ specifies that instances resulting from the query *qS* over the sources correspond to instances in the global schema represented by the query *qG* (similarly for an assertion of type ($qG \rightsquigarrow qS$).

3. THE DATA INTEGRATION PROBLEMS

Merging required data from different source is not an easy job, it has lot of problems like as Detecting correspondences between similar concepts that come from different sources, and conflict solving. Combining data coming from different data sources and providing the user with a unified vision of the Data.

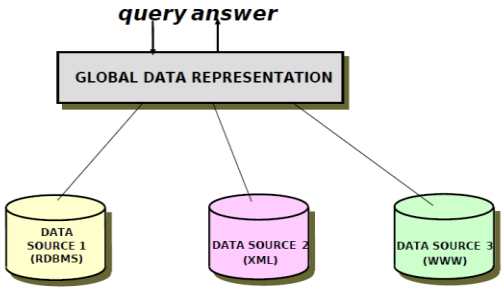

Figure 1: About Data Integration Problems

3.1 Problem of Designing Data Integration Systems

The problem of designing data integration systems is important in current real world applications, and is characterized by a

number of issues that are interesting from a theoretical point of view.The typical problems of data design are given a data model**,** organizing data according to the chosen data model in order to avoid inconsistencies and allow query optimization.

4. Data Integration in The General Context

For integrating Data we need to follow some steps: Source schema identification (when present)**,** Source schema reverse engineering (data source conceptual schemata)**,** Conceptual schemata integration and restructuring**,** Conceptual to logical translation (of the obtained global schema) and Mapping between the global logical schema and the single schemata (logical view definition). After integration: query-answering through data views.

Mapping between the global logical schema and the single source schemata (logical view definition) has two basic approaches - GAV (Global As View) , LAV (Local As View).

*GAV (Global As View***)** can be used also in case of different data models. In that case a model transformation is required GAV. Up to now we supposed that the global schema be derived from the integration process of the data source schemata. Thus the global schema is expressed in terms of the data source schemata. Such approach is called the Global As View approach.

In LAV (Local As View) The global schema has been designed independently of the data source schemata. The relationship (mapping) between sources and global schema is obtained by defining each data source as a view over the global schema.

GLAV (Global and Local As View) The relationship (mapping) between sources and global schema is obtained by defining a set of views, some over the global schema and some over the data sources.

Query processing in data integration requires a reformulation step: the query over the global schema has to be reformulated in terms of a set of queries over the sources. A main theme will be the strong relationship between query processing in data integration and the problem of query answering with incomplete information.

5. Ontology

Ontology is a controlled vocabulary that describes objects and the relationships between them in a formal way. It has a grammar for using the terms to express something meaningful within a specified domain of interest. The vocabulary is used to express queries and assertions. Ontological commitments are agreements to use the vocabulary in a consistent way for knowledge sharing.

Informally, we define an ontology as an intentional description of what's known about the essence of the entities in a particular domain of interest using abstractions, also called concepts and their relationships. An ontology must provide knowledge in the form of concise and unambiguous concepts and their meanings.

An ontology is (part of) a knowledge base, composed by:-
• a T-Box: contains all the concept and role definitions, and also contains all the axioms of our logical theory(e.g. "A father is a Man with a Child").
• an A-box: contains all the basic assertions (also known as ground facts) of the logical theory(e.g. "Tom is a father" is represented as Father(Tom)).

5.1 OWL (WEB ONTOLOGY LANGUAGE)

OWL is a language for defining and instantiating Web ontologies based on XML and RDF (Resource Description Framework). OWL is designed for use by applications that need to process the meaning of an information instead of just presenting that information to the user. OWL can be used to explicitly represent the meaning of terms in vocabularies and the relationships between those terms; by this language it is possible to infer new knowledge from a conceptualization using a specific software called reasoner. OWL provides three increasingly expressive sub-languages also called OWL dialects:-

OWL Lite: Provides classi cation hierarchies and simple constraints, it only permits to express relationships with maximum cardinality equal to O or 1.

OWL DL: Supports those users who want a high expressiveness while retaining computational completeness and decidability. OWL DL includes all OWL language constructs, but they can only be used under certain restrictions (i.e. a class cannot be an instance of another class). OWL DL is so named due to its correspondence with Description Logics (see below).
OWL Full: Provides the maximum expressiveness and the syntactic freedom of RDF with no guarantees on computational complexity. A key di erence of this dialect from the former is that a deductive process within such a theory can be undecidable.

An ontology as a schema integration support tool for content interpretation, wrapping, inconsistency detection and resolution.

5.2 Ontology Mapping

Ontology Mapping is the process of relating similar concepts or relations of two or more information sources using equivalence relations or order relations. These relations are commonly implemented in inference and reasoning softwares, so we can use the output ontology to perform complex tasks on them without extra effort.

5.3 Ontology Merging

Ontology Merging is the process of creating one ontology from two or more source ontologies with overlapping concepts or definitions. In the merging process the merged ontology is created from scratch, unifying all the source ontologies. In Ontology Merging there is no need for any reasoning software extensions because we reuse parts of sources ontologies without introducing new relations.

6. Ontology Integration

Ontology Integration is similar to Ontology Merging, but here the integrated ontology is created reusing parts of source ontologies as they are. A key task in Ontology Integration and Ontology Merging is the consistency checking that must ensure the absence of unforeseen or wrong implications into the merged ontology.

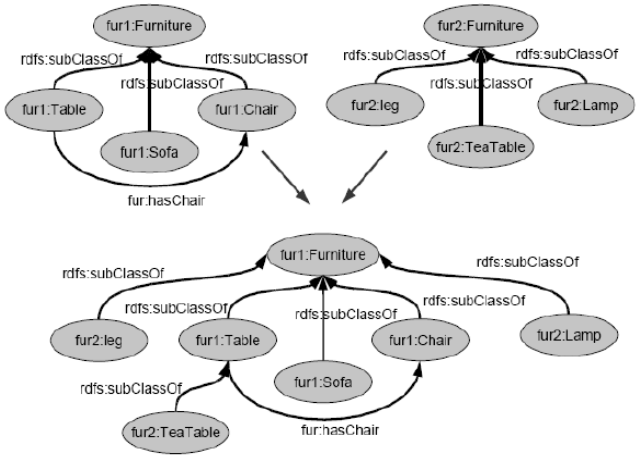

Figure 2: Ontology Integration

Integrated ontology used as a schema for querying, Ontologies used to represent the semantics of schema elements (if the schema exists).Similarities between the source ontologies guide conflict resolution.

6.1 Ontology Based Query Processing

All ontologies we are considering do not contain individuals which are instead managed by heterogeneous data sources. Description Logics has been defined to efficiently address the problem of query answering in Description Logics under constraints (exclusion dependencies and key constraints) , we check the model composed by the Domain ontology, Mapping ontologies and Data source ontologies for consistency and we classify it in order to provide a sound query rewriting algorithm. All considerations regarding constraints over the global schema and their satisfaction will be the subject of future work. We note that considering the ontologies automatically extracted by ROSEX (which use disjunctions between concepts and functional roles) the mapping language and RDFS as the language for expressing the Domain ontology, the formalism of the global model.

Semantic Query distribution is the process of distributing a query (specified over the global schema) towards the data sources, following mappings between the Domain ontology and the Data source ontologies. This problem has already been addressed in the data integration literature under the names Query rewriting and Query reformulation.

The first issue we have to address is to enrich the query using the Domain ontology in order to find all the applicable mappings. Informally, considering conjunctive ABox queries (queries composed only by a conjunction of class membership assertions and property membership assertions) all we have to do is to ask a reasoned for classification and to use the inferred model to expand the query.

After this expansion algorithm will have to handle GAV and LAV mappings. Informally ,using GAV mappings rewrite a term belonging to the Domain ontology (specified by the query) with a set of terms belonging to a Data source ontology while using LAV mappings rewrite a set of terms belonging to the Domain ontology with a single term belonging to a Data source ontology.

Indeed, because of the previous considerations regarding our environment, we address these issues following a bottom-up process, by taking into consideration the current state of the art in ontology-based data integration and trying to fit current leading technologies such as SPARQL and OWL to our needs. One important thing to note is that SPARQL is in fact an RDF query language, and that we have to rewrite SPARQL queries in terms of the data source structures following mappings, for these reasons we have restricted our mapping language in order to be able to rewrite the mappings into the SPARQL syntax. The system has to answer SPARQL queries expressed in terms of the Domain ontology. The problem is to parse the query and to "move" the query towards the Data sources.

6.2 Query Rewriting Using Gav Mappings

GAV mappings are usually resolved through unfolding, that is, by simply expanding the head of the mapping with the body of the mapping, in fact these mappings directly specify how to compute (virtual) instances of the Global schema using the data sources.

Input: A query Q' already expanded over the Domain Ontology, a query subgoal g and a GAV mapping m_i.

Output: Q'_{rew} a query where subgoal g is expanded with GAV mappings.
1: if isApplicable(m_i, g) then
2: Q'.addRewriting (g, body(m_i))
3: end if
4: return $Q'_{rew} = Q'$

Algorithm 1: GAVHandler (Q',g,v_i)

The problem of query rewriting using GAV mappings can be performed in deterministic polynomial time, since GAV mappings directly specify how the rewriting has to be performed.

6.3 Query Rewriting Using Lav Mappings

LAV mappings specify atoms belonging to DSOs as queries over the Global schema. Since a user specifies queries over the Global schema, an algorithm should infer the inverse views to rewrite a query q, originally expressed in terms of DO, in terms of Data source ontologies. This problem is well-known in data integration and query optimization and has been extensively studied.

The problem of finding all the answers to a query given a set of views is formalized by the notion of certain answers (the definition in distinguishes the case in which the view extensions are assumed to be complete (Closed-Word Assumption) from the case in which the views may be partial (Open-World)).
Informally, with the Closed-Word Assumption, we assume that the views are assumed to contain all the tuples that would result from applying the view definition to the database. Conversely, with the Open-World Assumption the extension of the views may be missing some tuples (but they may not have incorrect tuples.

Input: A query *Q'* already expanded over the Domain Ontology, a query subgoal *g* and a LAV mapping (view) m_i.

Output: Q'_{rew} a query where subgoal g is expanded with LAV mappings.

1: if isApplicable (m_i, g) then
2: Q'.addRewriting (g, head(m_i))
3: end if
4: return $Q'_{rew} = Q'$

Algorithm 2: LAVHandler (Q',g,v_i)

6.4 Combining Gav and Lav Mappings

GAVhandler and LAVhandler, which given a query subgoal g rewrite it in terms of data sources using mappings.

Input: A query Q and a set V of LAV and GAV mappings.

Output: Q'' a query where each subgoal is (possibly) expanded with mappings. Q'' is represented by an AND/OR tree where each AND/OR traversal represents a query rewriting.

1: Q'= DOhandler(Q);
2: for all Subgoal g_k such that g_k such that $g_k \in Q'$ do
3: for all Mapping (view) v_i such that $v_i \in V$ do
4: if GAV (v_i) then
5. Q'= **GAVHandler (Q',g_k,v_i);**
6. else if LAV(v_i) then
7. Q'= **LAVHandler (Q',g_k,v_i);**
8. end if
9. end for
10. end for
11. return Q''=Q'

Algorithm 3: queryExpand (Q,V)

7. Conclusions

The aim of this paper was to provide an overview of some of the theoretical issues underlying data integration. Several interesting problems remain open in each of the topics that we have discussed. For example, more investigation is needed for a deep understanding of the relationship between the LAV and the GAV approaches. Open problems remain on algorithms and complexity for view-based query pro cessing, in particular for the case of rich languages for semi structured data, for the case of exact views, and for the case of integrity constraints in the global schema. Query processing in GAV with constraints has been investigated only recently, and interesting classes of constraints have not been considered yet. The treatment of mutually inconsistent sources and the issue of reasoning on queries present many open research questions.

8. REFERENCES

i. ***Proceedings of the twenty-first ACM SIGMOD-SIGACT**: Maurizio Lenzerini - Data Integration: A Theoretical Perspective **(2002)***

ii. ***A Federated Ontology-Driven Query-Centric Approach:** Jaime A Reinoso Castillo, Adrian Silvescu, Doina Caragea, Jyotishman Pathak, Vasant G Honavar - Information Extraction and Integration from Heterogeneous, Distributed, Autonomous Information Sources .*

iii. *Chokri Ben Necib and Johann-Christoph Freytag -**Ontology based Query Processing in Database Management Systems***

iv. ***Technical report, ESWC2007**, Marcelo Arenas, Bijan Parsia, Claudio Gutierrez, Jorge Perz, Axel Polleres, and Andy Seaborne. Sparql - where are we? current state, theory and practice. **(2007)***

v. ***W3C recommendation,W3C,February2004:** Jeremy J. Carroll and Graham Klyne. Resource description framework (RDF): Concepts and abstract syntax..http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/.*

vi. ***In Proceedings of the ECAI 2000 Workshop on Applications of Ontologies and Problem-Solving Methods:** Berlin,Corcho, O. and Gomez-Perez, A. Evaluating knowledge representation and reasoning capabilities of ontology specification languages. **(2000)***

vii. ***IJAST Volume 34, September 2011:*** *Amir Fallahi and Shahram Jafari,- An Expert System for Detection of Breast Cancer Using Data Preprocessing and Bayesian Network, (2011)*

viii. ***IJAST Volume 35, October 2011****: K. Koteswara Rao, Srinivasan. Nagaraj, Dr GSVP Raju, - The Efficient way to Identify the Regular Expression in Text Databases (2011)*

ix. ***IJAST Volume 27, February 2011****: Tajunisha N, Saravanan V, A new approach to improve the clustering accuracy using informative genes for unsupervised microarray data sets*

x. ***ACM Transactions on Information Systems****: Goh, C.H., Bressan, S., Siegel, M. and Madnick, S. E. Context Interchange: New Features and Formalisms for the*

xi. *Intelligent Integration of Information. Vol. 17(3), 270–293, **(1999).***

xii. ***ICEIS:*** *Medcraft, P., Schiel, U., Baptista, P. DIA: Data Integration Using Agents. Databases and Information Systems Integration. 79-86, **(2003)***

xiii. *12.**Kluwer Academic Publishers, Boston.** Mena, E., Kashyap, V., Sheth, A. and Illarramendi, A. Observer: An approach for query processing in global information systems based on interoperation across pre-existing ontologies. http:, //citeseer.nj.nec.com/ mena96 observer.html, 1-49, **(2000)**.*

xiv. ***In Proceedings International Conference on Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems, Irvine CA,****- Nam, Y. & Wang, A. - Metadata Integration Assistant Generator for Heterogeneous Distributed Databases. 28-30. **(2002).***

xv. ***In Proceedings of the International Conference on Formal Ontology in Information Systems (FOIS'98), IOS Press:*** *Visser, P. R. S., Jones, D. M., Bench-Capon, T. J. M. and Shave, M. J. R. Assessing heterogeneity by*

xvi. *classifying ontology mismatches. 148–162, **(1998)***

xvii. ***Proceedings of IJCAI-01 Workshop: Ontologies and Information Sharing, Seattle, WA****,- Wache, H., Vögele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H. and Hübner, S. Ontologybased Integration of Information - A Survey of Existing Approaches 108-117, **(2001)**.*

xviii. ***Second International Conf. Parallel and Distributed Information Systems. January****, Woelk, D., P. Cannata, M. Huhns, W. Shen, and C. Tomlinson. Using Carnot for Enterprise Information Integration. 133-136, **(1993)**.*

xix. ***International Conference on Extending Datadase Technology:*** *Woelk, P., Kim, W. And Lee, W. Query Processing in Distributed ORION. 169-187, March, **(1990).***