Journal of Theoretical and Applied Information Technology
20th February 2014. Vol. 60 No.2
© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A CRITICAL OVERVIEW OF EXISTING QUERY PROCESSING SYSTEMS OVER HETEROGENEOUS DATA SOURCES

[1]Naoual MOUHNI, [2]Abderrafiaa EL KALAY
[1] University Cadi Ayyad, Faculty of sciences and Technologies
[2] University Cadi Ayyad, Faculty of sciences and Technologies
E-mail: [1] naoual.mouhni@edu.uca.ma , [2] a.elkalay@uca.ma

ABSTRACT

In the past, to answer a user query, we generally extract data from one centralized database or from multiple sources with the same structure. then things have been changed and we are facing the fact that in some cases, it is necessary to use a set of data sources to provide a complete information. these sources are physically separated, but they are logically seen as a single component to the final user. Besides the structure heterogeneity, there is another important point for what specialists are trying to find a solution which is the semantic heterogeneity of data sources. In this paper we are going to provide a list of different approaches that treated the query processing problem on heterogeneous data sources under different angles

Keywords: *Query Processing, Heterogeneous Data Sources, Ontology, Data Warehousing, Federation*

1. INTRODUCTION

Nowadays, with the great expansion of information within internet, and the use of multiple data sources which can be heterogeneous and physically separated, the old way used to process user queries is no longer the same, that's why databases specialist try to find new approaches to fill this gap.

the first step was by proposing many methodologies to integrate heterogeneous data sources, since in such situation, each independent database has its own schema, expressed in its own data model and have its own query language [19], so a user query must follow these steps to be executed:

- Resolving incompatibilities between databases in question by matching data types and attributes names;
- Resolving redundancy problems (same information in two places) ;
- A query language for the user who does not need to know the source query language or it's schema;
- Decomposing user's query into multiple sub queries to be executed on different databases sites;
- A function to integrate the results into one global result understandable by the final user.

We will discuss in section 2 the heterogeneous data integration problem, and several strategies for solving it. then in section 3, we will focus on the query processing in such heterogeneous environment , not only in distributed data sources which could be organized between them as a result of horizontal, vertical or other repartition approach, but over heterogeneous independent, autonomous data sources which make the querying process a complicated mission.
Then in section 4 we represent three examples of existing projects that treat the query processing problem across heterogeneous data sources.

2. INTEGRATION OF HETEROGENEOUS DATA SOURCES

Data sources integration techniques climbed to the next level well beyond traditional integration tools such as JDBC (Java Database connectivity) or ODBC (Object Database Connectivity) which connect relational databases together [20]. Now, the data may be stored in heterogeneous distributed or not distributed structures, even in the same scientific, economic or biology field which generate a huge amount of data that increase dramatically every day. the data

© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

manipulated inside the same scientific branch could be heterogeneous not only in structure level but also semantically. For example , if we take medical information all over distributed data sources, if they are not semantically integrated , we can find a medicine side effects expressed differently in two sites with two different expressions but in fact it refers the same thing. If these sources are integrated, then systems can answer user queries properly.

Several researches has been done to improve integration techniques, so we can cite the use of Data warehousing where data is extracted from diverse sources, transformed to be homogeneous by applying transformation functions like attributes form, e.g., date form yyyy-mm-dd into dd-mm-yyyy, and then loaded into structured Data warehouse.

actually, even by using Data warehousing , with the grow that some fields know, we find a structure with several heterogeneous data warehouses as treated in [22], hence the necessity to federate these components and use an ontology as we are going to explain in the section "Ontology based approaches" besides algorithms to integrate all the information sources.

So we can resume the impediment to data integration as follow:

– Structural heterogeneity: differences in type of attributes and its structure ;
– Semantic heterogeneity: differences in language used to represent the attribute ;
– Representational heterogeneity: difference in model and representation schema type that could be for e.g. relational, object oriented .

these elements push us to talk about two processes used to identify the issues above, which are matching and mapping.

the matching process is used to identify if two elements are semantically related or not, for example; if we take the two schemas DB1.Client (ID, Name, Class) and DB2.Customer(CID, FName, LName, PointNbr), 0the operation consists on matching the elements that are semantically related but differently represented , in this case we can cite ID in DB1.Client and the IDC in DB2.Costmer, which refer to the client identifier.

The mapping process is the transformation operations between elements, for example attributes class and pointNbr are used to classified customers so we can map it to class by assigning an interval of points to a class, so (0-1000; C) (1000-5000;B).

Data integration systems are quite different even if it is trying to solve the same problem, that why we are going to list some of the important approaches in this domain.

A centric data integration architecture also called central data integration, is a model of system that has a global schema, which provide the final user with a uniform interface to access information stored in heterogeneous data sources by means of queries using the terms of the global schema [31],[5],[17].

On contrary to Centric architecture, in the peer-to-peer data integration system , any peer or data source can accept user queries to access in other peers[31],[17].

2.1 Mapping Approaches And Query Processing

As we said above, to ensure data integration, one of the main operations to be done is the mapping, even in centric or peer-to-peer data integration systems.

In central data integration case, the mapping is established between the global schema and the data sources schemas., While in peer-to-peer data integration systems, mapping is created between peers (data sources).

There is two common approaches to create this types of mapping; Global as View or GaV and Local as view or LaV [31],[17],[20].In the GaV approach, we associate a view over the data source w3ith every entity in the global schema, which facilitate querying because the mapping is explicitly defined. While by using the LaV approach the query processing is more complex since the local schemas are defined as views over the global schema.

And since in every system, it is necessary to take by consideration an update strategy, in GaV approach every time there is a change to the data sources the views are changed. however, the LaV allows for changes to the data sources that do not affect the global schema.

The LaV is a view based approach, which could make query processing a complicated mission, indeed, in case of the LaV approach, user can't access data source relations , the only information that he dispose is through the views .

However, the GaV approach seems to be more easier to process queries, since the mapping specifies directly which element of source queries is corresponding to which element of the global schema.

© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2.2 Ontology Based Approaches

Several methodologies has been developed to solve the integration issue, one of them is the use of ontology, which founded in the field of philosophy. An ontology is in the basis of semantic data integration process, it is a way of using a conceptual representation of data and of their relationship to eliminate heterogeneities, it is defined as an explicit specification of a shared conceptualization [31].

Three approaches are found in using ontology in data integration process, as shown in the figures bellow :

Fig. 1. Single Ontology Approach

Fig. 2. Multiple Ontology Approach

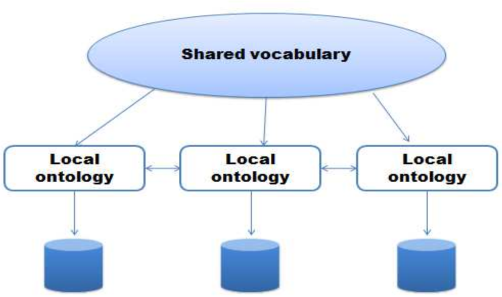

Fig. 3. Hybrid Ontology Approach

The first approach (Fig. 1), is based on the usage of one shared ontology over all the data sources. it seems to be the simplest approach in comparison with the others. however, this approach has its disadvantage in its simplicity, since in some cases we need detail specification rather that a global ontology that treat all the sources as one view if they are integrated together.

The second one is the Multiple ontology approach (Fig. 2), it is characterized by the usage of local ontology for each data source. Every data source has his own ontology and its integrated in harmony with the others. the problem in this case, is that by using many local ontologies we are facing the same problem as the first one, if they are not sharing a common vocabulary, we need to define ontology for the set of local ontologies.

The last approach is called hybrid ontology approach (Fig. 3), this one seems to be the more convenient for big projects, in which data sources may be extremely different , so it propose to use a shared vocabulary to integrate local ontologies that are defined on each data source.

3. QUERY PROCESSING ACROSS HETEROGENEOUS DATA SOURCES

Data integration can be described as a set of independent, heterogeneous data sources covering the same domain of interest.

The spinal cord of data integration process, is the query processing, as we saw above, one of the points that must be present in a data integration system is a common query language that must be defined despite all the data sources query languages.

3.1. Query model

Every data source has his own query model, which is the model of data storage that must be known by the final user who wants to execute query on this site. the query model is characterized by four components [30], the first one is the abstract model which gives an idea on the type of data structure that can be handled by the data source e.g. text files, hierarchical database or relational tables). The second component is the schema of data which specify the representation and localization of data in the database, if the user query is about two elements a and b, are a and b specified in the same file ? or it is necessary to join 2 files?. query language, which will be used to interrogate the database (e.g. SQL), and the last component is the format of the database data; we are going to explain this component via the example of the two elements seen in section 2; DB1.Client (ID, Name, Class) and DB2.Customer(CID, FName, LName, PointNbr), for these data sources (DB1 and DB2) the customer name is stored differently, so we need to know before creating the query how this data attribute is stored .

Journal of Theoretical and Applied Information Technology
20th February 2014. Vol. 60 No.2
© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Generally in an heterogeneous data sources integration system we need to represent these elements to insure the integration process. in (Fig. 4) describes a model of standard architecture for such system.

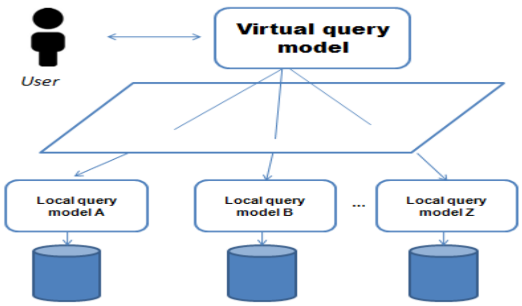

Fig. 4. Position Of The Virtual Query Model In A Heterogeneous Data Integration System

The heterogeneous data sources integration system offers generally a virtual query model that allows to the final user to access data sources without knowing about its local query model, all the process should be transparent to the user.

Since the system uses a global query model, it is necessary to define the four components of this global query model, in the next section we are going to cite a list of the important solutions proposed in relation with this issue.

3.2. Query model vs. ontology

An ontology may resembles to a query model if it includes these items:

- A formal abstract model for representing the properties of objects in a domain ;
- The schema component of query model: A definition of the objects classes and of the relations and functions that may be defined over the members of those classes in a particular domain ;
- The format component of a query model: A specification of the object constants that may be members of the defined object classes.

And it will be exactly the same think as a query model if it involves a query language [9].

3.3. Query Languages

As we have seen previously, in a query processing across an heterogeneous systems, the original query will be executed over multiple data sources that may have each one a different query language. Thus, the first question to be asked is ; does the data source require low-level language which gives details on target organization, or it require a declarative high-level language that specifies what to retrieve, rather than how to retrieve it ?

By analyzing several works on heterogeneous data sources integration's systems, we can list a set of languages uses to formulate user's query, as an example of XML sources Integration ; [6] uses a declarative query language OQL modeled after SQL and have similarities whit it, the principle of this approach is to query the database via the global schema, using simple query tree, based on Select, from and where clauses.

Other experimental Heterogeneous data bases systems have used a simple set oriented query language such as *DAPLEX* [29].

The *XQuery* language is a W3C emerging standard that is used for querying XML sources, many applications such as XML EDI and XML portal applications, are using the wrapper to transform relational source in an XML data source supporting *XQuery* commercialized under the name *XMLizer* [12].

In addition to *XQuery* and *OQL* languages, there is another tool used to interrogate xml heterogeneous sources called Quilt; which unifies the once separate world of documents and relational databases. here you can see an example of *Quilt* syntax:

```
/* List all the titles with the
word "data" */
<queryresult>
      FOR $title IN
      document("book.xml")//title
      WHERE
      contains($title,"Data")
      RETURN
<datatitle>
      $title/text()
</datatitle>
</queryresult>
```

This part of code, illustrates a *Quilt* query to extract all the titles which contains the substring "data" [23].

In other context; heterogeneous data warehouses Integration systems, in [28] a formal framework for data integration across an heterogeneous federated data warehouses system has been presented, the query language used for this application is *SQL*.

As you can see, different languages have been used for the query processing; in XML data sources , relational data bases or federated data warehouses. To chose one over the other you have to take in

Journal of Theoretical and Applied Information Technology
20th February 2014. Vol. 60 No.2
© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

account the data sources capabilities and the system organization.

3.4. Query Translating, Rewriting And Loss Of Information Measurement

The second step in query processing is to translate and rewrite the user query formulated in one of the languages discussed above , to equivalent queries formulated in specific data sources language and in target specific terms. The element of the data integration system that insure this task is called *mediator*, *driver* or *encapsulator* [30],[2].
this element is the implementation of one of the rewriting algorithms such as *the Bucket* algorithm introduced in [14] which use the sources views to reformulate conjunctive queries expressed in global schema terms, as an extension of this algorithm [27] has introduced the *MiniCon* algorithm which exploits input/output dependencies between the query data sources for reducing the search space of possible rewritings.
Another algorithm that resembles to the *MiniCon* is presented in [6] to querying XML heterogeneous sources using ontology based mediator which exploits the parent/child dependencies of query variables for query decomposition.
Some of these drivers are following GaV approach, such as *YAT* [7], *Tsimmis* [24] and *MIX*[4], when others are following the LaV approach like *Information Manifold* [14], *Tukwila*[27].
However, several mediators such as *Information Manifold*, *SIMS*[32] and *Infosleuth* [26] are providing query processing using concepts but it still modest, in order that they are using ontologies to describe declarative specification of semantic information and forgot about the syntactic data representation. while other approaches like the *OBSERVER* which uses multiple ontologies , and gives the user the possibility to choose a convenient ontology to express its query according to his domain and context, then the system takes the responsibility to rewrite this query in local ontologies language, they are using a system based on Description logic to express the ontologies used to describe data repositories content.
The query processing approaches that exists, are decomposing user query into several sub queries in order to be executed on different targets called *sub goals.*
Some of them are using methods that performs translation using synonym, hyponyms and hypernyms relationships (e.g. The OBSERVER), and analyze data repositories query processing capabilities to prepare the adequate *wrapper* in order to improve the query processing.

A wrapper is a module that knows the data organization, extract data from data sources and hide the structure to other component of the information system [16].
We can classify the existing approaches into two classes as seen in the figure (**Fig. 5**) thereafter:

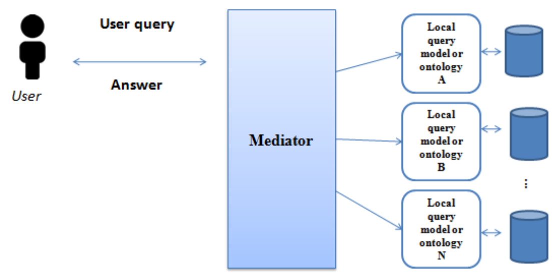

Fig. 5. Mediator Based Approaches

as we have seen above, in mediator based approaches, the user is not implicated in the choice of the appropriate ontology (if we are in an ontology based environment), the mediator take the responsibility to find the adequate data repository and translates the user query into data source's query language. In this case the user is dependent to the encapsulator/mediator.

In other hand, there are another systems which involves the user to improve the query processing. these systems are characterized by the use of descriptive languages in the side of user and make possible during the user query formulation to choose the appropriate term that satisfy user need.
Generally the system structure resembles to the figure (**Fig. 6**) below:

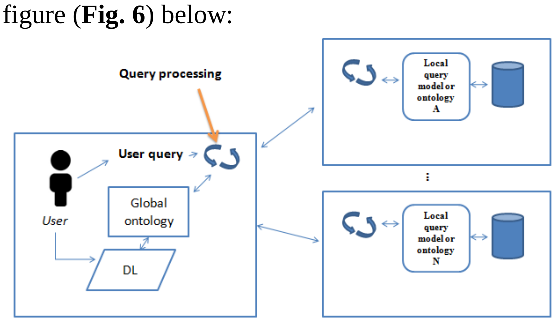

Fig. 6. Non Mediator Based Approaches

Thereby, the second type of data integration systems are convenient to big projects rather than first type's systems, since there is no Descriptive Language, the process may be less performance.
Another important point in query processing, is the measurement of loss of information during query processing. In ontology based Integration systems for example, after translation of user query, we can

Journal of Theoretical and Applied Information Technology
20th February 2014. Vol. 60 No.2
© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

have terms without equivalent in existing ontologies. In this case as Eduard Mena has cited in his paper [9] : " *Each conflicting term in the user query is then replaced by the intersection of its immediate parents or by the union of its immediate children. This method is applied recursively until a translation of the conflicting term is obtained using only the terms of the target ontology*".

This solution, could have a negative side where she could change de semantic of the user query's answer.

Some works are defining an allowed loss of information percentage (0 to 100) during the user query definition phase. so the terms of the original query will be replaced by the target ontology terms with respect to the percentage of information's loss .

In literature, several approaches were implemented to compute and approximate the user query answering loss of information in case there are multiple answers from multiple data sources. Some projects like the Multiplex project [1] are trying to measure divergence from the true answer basing on modeling; they are using intersection and unions of the candidate results to approximate the completeness and the soundness of the results. Others are using precision and recall estimation based on the sizes of the extensions of the terms to evaluate a numeric measure that represents the loss of information [9].

In other hand, some possibilistic approaches are used in works like [25] and [8], in [25] for example, they are querying uncertain data using numerical probabilistic to estimate the loss measures.

this measure is what they will use to fix the percentage of information's loss as seen previously .

4. EXAMPLE OF EXISTING ARCHITECTURES

Numerous works have been developed to cover the integration of heterogeneous sources issue. We chose to represent three of those projects ; the raison behind this choice is that we want to represent an example for different domain and data sources structure (e.g. XML data sources types, Relational database and Data warehouses).

4.1. An Xquery-Based Query Processing System

In recent years, Numerous projects have been focusing on developing Mediator based applications to integrate heterogeneous data sources and query processing (e.g. Garlic [13], IRO-DB [11],...) those projects are using a driver / mediator which provide a uniform user interface to query integrated views of heterogeneous information sources. Also, they are using Wrappers to provide local views of data sources in a uniform data model [12].

After that, the focus has come to XML as a tool for exchanging and representing data in a simple way. One of the projects developed for XML heterogeneous data sources is the XQuery-based Integration system .

This project is based on a mediator called e-XMLMedia, this mediator is using XQuery (described previously in section 3.3.) as an XML query language .

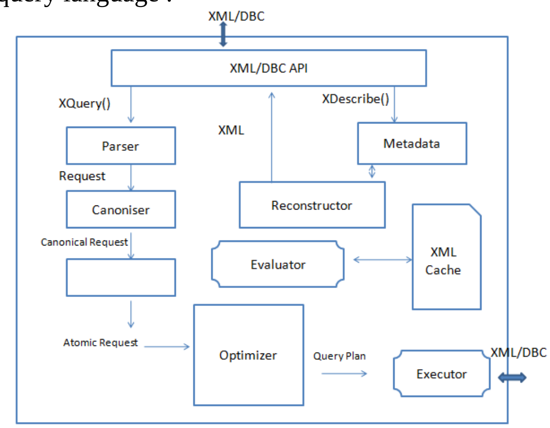

Fig. 7. XQUERY Mediator Architecture

As described in (Fig. 7), the e-XML Mediator is containing, several components that decompose the user query into multiple sub-queries (mono-source queries), and getting results in XML through a SAX interface, then assembling them.

The Mediator is based on an XML/DBC API which is the interface between the Mediator and the exterior, user query is written in XQuery language ,then PARSER controls if the query is syntactically correct otherwise, a document error is generated. after that the correct query is normalized by the CANONISER and turns into simple query. The DECOMPOSER, create multiple sub-queries , the execution plan of those queries is generated by the OPTIMIZER that localizes data sources using Metadata repositories. results are then post evaluated to form the global answer.

4.2. OBSERVER : An Architecture To Support Query Processing

OBSERVER is one of the non Mediator solutions, that used ontologies to solve the data heterogeneity and query processing problem. the aim of this solution as seen in (Fig. 8.) is to use ontology for integrating heterogeneous sources and query processing without loss of information.

© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

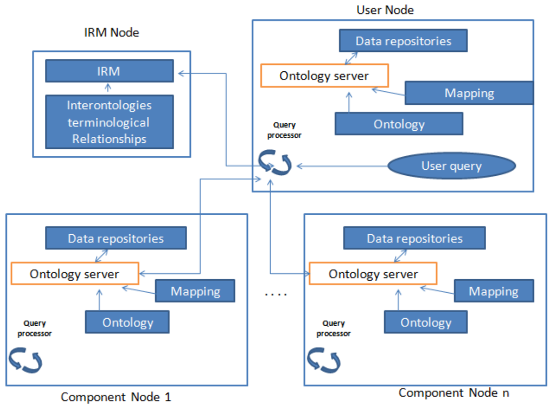

Fig. 8. OBSERVER Architecture

The three steps for query processing in OBSERVER system; in query construction step, the user choose the ontology that fit to his objectives, the percentage of loss of information , then in step 2, the query processor invokes the ontology server to detect the target data source, in step 3, is a controlled and incremental query expansion to new ontologies process.

This system is provided by an InterOntology Relationship Manager (IRM) , which support ontology-based interoperability and solve the shared vocabulary problem [9].

4.3. Ontology Based Data Warehouses Federation Management System

Data warehousing is in its self a solution for heterogeneous data Integration problem. it allows to integrate data from multiple heterogeneous data sources, transform it and then load it into a structured data warehouse.

The problem persists when we are in presence of many data warehouses that are heterogeneous , this project is about the amelioration of existing algorithms to integrate federated data warehouses using ontology.

this schema resumes the structure proposed by [22]:

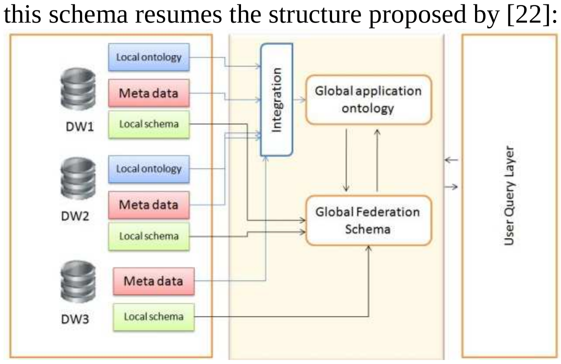

Fig. 9. Data Warehouses Federation Management System Using Ontology

Every component, may have his own local ontology (or not), these local ontologies are integrated into one global ontology used in query processing and in the construction of the global federation schema. This system is using SQL as a high level query language to formulate user query.

5. DISCUSSION AND CONCLUSION

With the growth need for integrating heterogeneous data sources, that exists independently all over the world, autonomous but connected to provide a complete information to the final user. Many applications have been developed for solving this issue taking by consideration the variety of data sources that exists (XML data sources, relational data bases , oriented object data bases, data warehouses,...), and providing to the user an interface that allow to formulate queries independently of the data localization or structure.

Those systems solved the tackle issue in different ways, by using ontlogy, simple taxonomies to overcome the semantic heterogeneities. Others are presenting Mediators and wrappers for processing queries using sophisticated algorithms and technologies.

The heterogeneity problem still persists, due to the dynamic nature of the data sources, if we take for example the factor of big amount of data inserted via social networks , e-commerce and even in big service companies and laboratories, we are facing the fact that data despite the structure where it is stored(big data or integrated into data warehouses) suffer from the problem of dirty data and that still in even after the data cleaning process, which can influence the decision making in case of DW and a wrong answers to query user due to the presence of outlier sometimes or redundant data in other time. So, new track of research have been discovered, to optimize and improve existing systems and algorithms for data cleaning and query processing to meet the new data environment requirements; such as handling big amount of data in a efficient way, integrate data from different data domains and eliminating duplicated data in case of big datasets (semantic and syntactic redundancy) with the minimum lost of information by improving the learning machine algorithms and minimizing the human intervention . thus, this domain still need continuous and more studies to be done.

REFRENCES:

© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[1] A.Motro, "Multiplex: A formalmodel ofmultidatabases and its implementations", Technical Report, Technical Report ISSE-TR-95-103, Department of Information and Software Systems Engineering, George Mason University, Fairfax, Virginia, March 1995.
[2] All, A.R.a.D.L., QUERY TRANSFORMATION IN HETEROGENEOUS DISTRIBUTED DATABASE SYSTEMS. Computers ind. Engng 1996. Vol. 31(No. 1/2): p. pp. 323 - 326.
[3] Alon Y.Levy, A.R., Joann J. Ordille, Querying Heterogeneous Information Sources Using Source Descriptions. 1996.
[4] BARU C., GUPTA A., LUDÄSCHER B., MARCIANO R., PAPAKONSTANTINOU Y.,VELIKHOV P., CHU V., « XML-based information mediation withMIX », Demonstrations,ACM/SIGMOD, 1999, p. 597–599
[5] Bernd Amann , C.B., Irini Fundulaki, and Michel Scholl, Ontology-Based Integration of XML Web Resources.
[6] Bernd Amann, C.B., Irini Fundulaki, Michel Scholl, Querying XML Sources Using an Ontology-based Mediator.
[7] CHRISTOPHIDES V., CLUET S., SIMEON J., « On Wrapping Query Languages and Efficient XML Integration », Proc. of ACM SIGMOD, Dallas, USA, May 2000.
[8] D. Dubois, J. Lang, and H. Prade, "Automated reasoning using possibilistic logic: Semantics, belief revision, and variable certainty weights", IEEE Transactions on Knowledge and Data Engineering, vol. 6, no. 1, Ferbruary 1994.
[9] EDUARDO MENA, A.I., VIPUL KASHYAP, AMIT P. SHETH, OBSERVER: An Approach for Query Processing in Global Information Systems Based on Interoperation Across Pre-Existing Ontologies. Distributed and Parallel Databases, 2000. 8: p. 223–271.
[10] Elien Paret , W.V.W., Sven Casteleynb, Beat Signer, Olga De Troyer, Efficient Querying of Distributed RDF Sources in Mobile Settings based on a Source Index Model Procedia Computer Science 2011. 5: p. 554–561.
[11] Fankhauser P., Gardarin G., Lopez M., Muñoz J., Tomasic A.: "Experiences in Federated Databases: From IRO-DB to MIRO-Web", 24rd Very Large Data Bases, pp. 655-658, August 24-27, 1998, New York City, New York, USA, 1998
[12] G. Gardarin, A.M., T. Tuyet Dang-Ngoc, L. Smit, Integrating Heterogeneous Data Sources with XML and XQuery. Second International Workshop on Electronicy Business Hubs - WEBH (DEXA Workshop, Aix-en-Provence : France 2002.
[13] Haas L., Kossman D., Wimmers E., Yang J.: "Optimizing Queries across Diverse Data Sources", 23rd Very Large DataBases, August 1998, Athens, Greece, 1997.
[14] Halevy, A.Y., Answering queries using views: A survey. The VLDB Journal 2001. 10: p. 270–294.
[15] Ioana Manolescu, D.F., Donald Kossmann, Answering XML Queries over Heterogeneous Data sources. 1996.
[16] J. Hammer, M. Breunig, H. Garcia-Molina, S. Nestorov, V. Vassalos, and R. Yerneni, "Template-based wrappers in the TSIMMIS system", in Proceedings of the Twenty-Sixth SIGMOD International Conference on Management of Data, Tucson, Arizona, May 1997.
[17] Juan E., Ontology data integration for competitive decision making , 2010
[18] Lenzerini, M., Data Integration: A Theoretical Perspective.
[19] Mahendar MacUaavaram, D.L.A., Ming Zhou, INTEGRATING HETEROGENEOUS DISTRIBUTED DATABASE SYSTEM Computers ind. Engng 1996. Vol. 31(No. 1/2): p. pp. 315 -318
[20] Mike Nolen, L.M., Integration of Heterogeneous Data Sources using Ontologies: An overview of current methodologies
[21] Mitra, P., An Algorithm for Answering Quering Efficiently Using Views. 1999.
[22] MOUHNI N., EL KALAY A., Ontology based data warehouses federation management system International Journal of Computer Science Issues, vol 10, Issue 4, No 1, July 2013, p. 276-281.
[23] Orlov, M., Summary for Quilt, November 2000.
[24] PAPAKONSTANTINOU Y., GARCIA-MOLINA H., WIDOM J., « Object Exchange Across Heterogeneous Information Sources », Proc. ICDE, Taipei, Taiwan, March 1995,p. 251-260.
[25] Pauray S.M. Tsai and Arbee L.P. Chen, "Querying uncertain data in heterogeneous databases", in Third International Workshop on Research Issues in Data Engineering: Interoperability in Multidatabase Systems,Vienna, Austria, April 1993.

© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[26] R. Bayardo, W. Bohrer, R. Brice, A. Cichocki, G. Fowler, A. Helai, V. Kashyap, T. Ksiezyk, G. Martin, M. Nodine, M. Rashid, M. Rusinkiewicz, R. Shea, C. Unnikrishnan, A. Unruh, and D. Woelk, "InfoSleuth: Agent-based semantic integration of information in open and dynamic environments", in Proceedings of the 1997 ACM International Conference on the Management of Data (SIGMOD), Tucson, Arizona., May 1997.
[27] Rachel Pottinger, A.L., A Scalable Algorithm for Answering Queries Using Views, in Proceedings of the 26th VLDB Conference2000.
[28] Rafał Kern, T.S., Ngoc Thanh Nguyen, A formal framework for query decomposition and knowledge integration in data warehouse federations. Expert Systems with Applications 2013. 40: p. 2592–2606.
[29] Smith JM, Bernstein PA, Dayal NG, Landers T, Lin KWT Wong E.Multibas integrating heterogeneous distributed database systems. In: Gupta A, editor. Integration of information systems bridging heterogeneous databases. New York: IEEE Press, 1986 163–75.
[30] Sujansky, W., Heterogeneous Database Integration in Biomedicine. Journal of Biomedical Informatics, 2001. 34: p. 285–298.
[31] Xiao, I.F.C.a.H., Ontology Driven Data Integration in Heterogeneous Networks. ADVIS Lab Department of Computer Science University of Illinois at Chicago, USA.
[32] Y. Arens, C.A. Knoblock, and W. Shen, "Query reformulation for dynamic information integration", Journal of Intelligent Information Systems, vol. 6, nos. 2/3, pp. 99–130, 1996.