A Data Integration Approach Based on Indexation

Shokooh Kermanshahani[†]**, Hamid Reza Hamidi** [††],
Computer Engineering Department, Faculty of Engineering and Technology,
Imam-Khomeini International University, Qazvin, Iran

Summary
Information integration is the problem of combining and querying data from several autonomous and heterogeneous sources in a homogeneous fashion. There are many research projects that present different approaches. Depending on the integrated view, these approaches can be categorized into two main categories: materialized and virtual approaches; there are also some hybrid approaches when there is a composition of materialized and virtual views. The main advantage of a hybrid approach is to offer a trade-off between the query response time and data freshness in a data integration system. In the existing approaches, query optimization is often privileged for the materialized part of the system.

In this paper, we develop a hybrid approach which aims to extend query optimization to all the queries of the integration system. It also provides a flexible data refreshing mechanism in order to tolerate different characteristics of sources and their data. This approach is based on the Osiris object indexing system. Its indexation system relies on the partitioning of the object space using the view constraints.

Our hybrid approach, materializes the indexation structure of the underlying objects at the mediator level and offers more flexibility in data refreshing than a fully materialized approach and a better query response time in comparison with a fully virtual data integration system.

Key words:
Data Integration, Heterogeneity, Data warehouse, Mediator, Hybrid approach, Views.

1. Introduction

The first generation of databases and information systems was developed for single usage but with the growing importance of computer information systems, sharing and combining these sources has become inevitable. Most enterprises or organizations need to share and combine their data. ”The goal of information integration is to enable the rapid development of new applications requiring information from multiple sources” [1]. Several technologies have been developed for this purpose in the research domain as well as in practice. Two main types of integration have been proposed: integration of functionalities and integration of data [1, 2, 3].

In data integration, an integration system satisfies the interest of the users by integrating data of different sources under a sharing (common) semantic so that the integration system appears as an independent information source. To achieve this goal many challenges must be overcome; the autonomy, heterogeneity and interoperability of data sources must be considered and the inconsistency of data has to be solved [1, 2, 3].

One of the most important challenges for integrating different autonomous data sources is the heterogeneity which can appear at different levels. The hard ware on which two information sources are developed, the network protocols, the software, the data and the query languages may be different. However, the essential and more complicated aspect of heterogeneity is semantic heterogeneity. Semantic heterogeneity characterizes the differences in signification, interpretation or utilization of the same data [4, 5].

In a data integration system, data from different sources are integrated into a global integrated schema which satisfies the needs of users and which is managed by a management system. All heterogeneities are hidden from the user, who queries the global schema as a single database schema.

1.1 Our Contributions

Our first effort was to develop a fully virtual data integration framework based on a hierarchy of views [9]. We proposed a multi-mediator architecture in which each mediator corresponds to a view. Data sources are classified under the corresponding mediators. A user query is sent to a mediator according to the view under which it is classified. This classification is made by using the constraints of the views. Classifying a query under a view can reduce the search space for the query. It is based on the Osiris platform [10, 11, 12] which is a prototype of an object-based database and knowledge base system.

While studying this solution we found that we could obtain more advantages and a higher degree of semantic query optimization by profiting from the instance classification mechanism of Osiris which is based on view constraints in Osiris, hence our work on a hybrid approach to data integration, which we present now. The main idea of this paper is to develop a semi-materialized data integration framework, which represents a new aspect of a hybrid method.

Manuscript received July 5, 2017
Manuscript revised July 20, 2017

2. IXIA: A hybrid data integration

The existing hybrid approaches provide a rapid access to the materialized data. Other data remain in the local sources and are queried directly from the sources when necessary. As a consequence, only the queries to the materialized part of the system are optimized. A typical example of integration scenarios compatible with such approaches is the integration of geographical data, hotel and tourism information and weather information for a travel agency. In this example, the data of geographical and tourism centers are stable and can be materialized while other information such as weather data change more frequently and are integrated in a virtual manner. Many other data integration scenarios can profit from the trade-off that a hybrid approach offers between query response time and data freshness.

In this section, we present IXIA (IndeX-based Integration Approach) a partially materialized (hybrid) framework for a data integration system. It provides a query optimization to the integration system as well as a flexibility of data refreshing for different data sources, according to the needs of the integration application.

2.1 IXIA Architecture

Like a mediator approach, IXIA has a mediator-wrapper architecture, although with some materialization. IXIA has been developed based on the Osiris system in order to take advantage of its object indexation system. Osiris is an object-based database and knowledge base system based on a hierarchy of views where views are similar to concepts defined by logical properties, like in a Description Logic approach [13].

2.1.1 The P-type concept of Osiris

The Osiris system implements the P-type data model. The concept of P-type has been created in 1984 by Ana Simonet [14] with two principal objectives: data sharing through the views and automatic verification of integrity constraints. The P-types data model, where p stands for the French word “Partage”, which means shared, had been first designed in a database management perspective but it later proved to be adapted as well to knowledge base needs. Compared with other data and knowledge representation models, the novel characteristic of P-types is that views, which define a point of view on a family of objects, constitute its central concept: a p-type is not defined first, and then its views, but a P-type is defined through its views. An object is an instance of one and only one p-type, but it can belong to several of its views and change the views it belongs to during its lifetime. Classifying an object into the views of its p-type is a characteristic inherent to this model. This is why the Osiris system, which implements the p-type model, can offer functionalities for decision support, alert management, semantic query optimization, etc. In our work, we went deeper into the use of the p-type concept with the purpose of profiting from its object indexation system to develop an indexed-based data integration system.

As mentioned above, the main materialized part of the IXIA is the indexation structure which is based on the instance classification of Osiris. A direct advantage of this materialization is query optimization for the integration system.

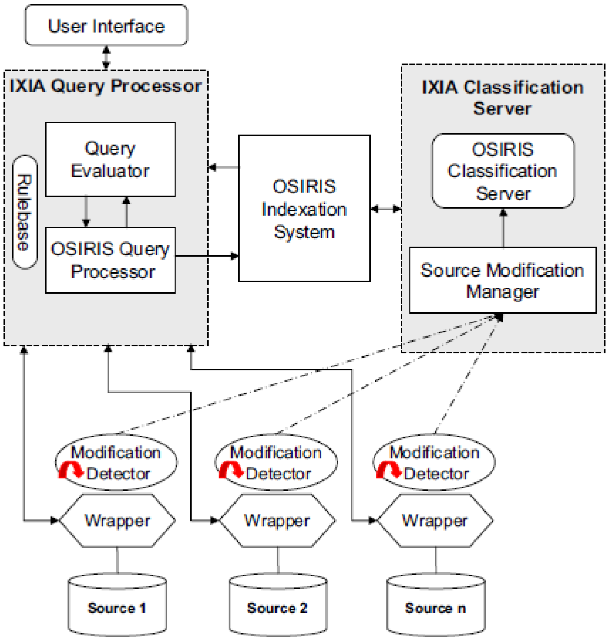

Figure 1: IXIA data integration architecture.

Figure 1 shows a presentation of the IXIA architecture. Only the relevant modules of Osiris are shown here. We briefly describe this architecture in the following two subsections:

2.1.2 Indexation and Indexation Maintenance

After defining the integrated schema (an Osiris schema), the classification server makes a first object indexation for all the sources objects which correspond to the global schema and sends the indexation data to be saved in the Osiris indexation module.

The indexation data are then incrementally updated by the classification server. The ”Modification Detector” modules detect if there is some updating in the sources which results in updating the indexation data from the last indexation maintenance. The Modification Detector of each source functions independently and can be executed with different frequencies.

Updating information obtained from the modification detectors is sent to the ”Source Modification Manager” module of the IXIA classification server. This module adds the source information and prepares the ”indexation repairing message” for the ”Osiris Classification Server”, which does the indexation maintenance just as in a single Osiris database.

We note that mappings between the object indexation and data in local sources are made in the wrappers. We save the (Oid, primary-key) correspondence between the Osiris objects of the global schema and the data in sources. Wrappers also do the mapping between the local sources' schemas and the Osiris Global Schema.

2.1.3 Query Processing

Query processing in Osiris and consequently in IXIA is done using the indexation information and provides a query optimization.

Depending on the method of schema transformation (LAV [6], GAV [7], BAV[8]), a query decomposition / reformulation is made in a virtual approach; query processing is done at the source level, then a mediator composes the partial responses with respect to data consistency and sends a unified response to the user. These processes are time-consuming. In most mediation approaches, the procedures involved in the query evaluation process are executed at query time.

In IXIA, some procedures associated with query evaluation are executed in an off-line manner, thus reducing the query response time. We call such procedures pre-procedures, and they consist of the indexation process and maintenance. In other words, in IXIA the problem of analyzing and processing a query is transformed into the problem of object indexing, refreshing this indexation, and searching the response objects' attributes in the sources. The query decomposition and the generation of the execution plan are done by the ”Query Evaluator” module of the IXIA query processor. The partial queries are sent to the Osiris Query Processor to find the satisfied objects using the object indexation system. Re-composition of partial responses into a final response is also done by IXIA Query Evaluator. The object indexation system of Osiris also takes advantage of the hierarchy of views of Osiris in its structure. This implies reducing of the search space at the mediator level in our integration approach.

A user query in IXIA is an Osiris query that the general form of it is:

(Context | conditions) [attributes]

Two scenarios are possible for this query:

1. Both *context* and *conditions* correspond to a single P-type of the global schema. The query is sent to the Osiris Query Processor in order to extract the response Oids.
2. The *conditions* of the user query correspond to more than one P-type of the global schema (two P-types, for example). In this case, the user query must be decomposed into several Osiris queries (each corresponding to a P-type). This decomposition is necessary because when a query corresponds to more than one P-type, we may need the value of one or several attributes in one P-type in order to query other attributes in another P-type. In the context of data integration, attributes are in different sources. Thus, often a part of the query may need the result of another part.

For each Osiris partial query, the Osiris Query Processor (OQP) searches the valid and potential Oids, which are Global Oids (Goid). The Query Evaluator prepares the source partial queries and sends them to the wrappers to verify all the complementary conditions and extract attributes. In this process, before sending partial queries to the sources, Goids are decoded in order to obtain the Loids (Local Oids) and the corresponding sources. We note that some attributes found in the materialized part of the Modification Detector may be used in the query processing. The partial responses will be re-composed by the IXIA query evaluator into an Osiris Response (a partial Osiris query). The final response for the user is prepared after receiving all the partial Osiris responses. In IXIA we then use the Osiris Query Processor only in single P-type query option.

The algorithm which must be followed by the IXIA query processor in order to make the decomposition, evaluation and re-composition of a user query is generated by the query evaluator using the sources information. It is memorized in the Query Plans module until the end of the processing of a user query.

3. The Motivation Application

Qazvin Telecommunications Company (QTC) is an Iranian regional telecommunications company covering ten cities. Each city contains several (up to 6) PSTN[1] switches depending on the number of subscribers. The total capacity of all switches installed in Qazvin province is about 430.000 subscribers.

[1] The Public Switched Telephone Network.

Qazvin OMC project[1] aims to create an operation and maintenance center in the Qazvin province so that QTC employees can operate and maintain all the PSTN switches. The platform has to connect remotely to the switches, manage them and access their data with the best possible delay. The user interfaces are expected to be uniform for all switches [15].

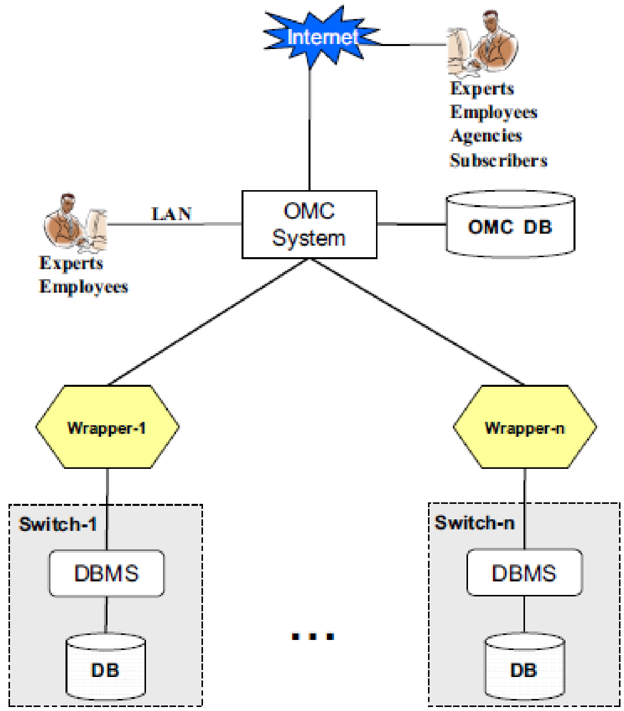

Figure 2: Designed platform of Qazvin OMC project.

Figure 2 demonstrates our designed platform in the Qazvin OMC project. It aims at reducing communications and remote operations on the switches. For this purpose, the OMC database keeps out the extracted information which is valid and does not need to be refreshed. This historical data will be integrated into a materialized system (data warehouse).

Figure 3 makes a global presentation of the telecom data integration system. It consists of a Data Warehouse (DW) for the historical data and a mediator system. Three groups of the data are saved in the DW. These are the data that change never:

1. In the billing system, the call details data are extracted from the switch databases and saved in the DW. These data are extracted at the end of each billing period.

   The switches insert a line in the log file when they dump these data from their modules to their database.

2. At the end of each billing period (e.g. at the end of each month) the bill data are saved in the DW.

3. Once an alarm is recovered it is saved in the DW.

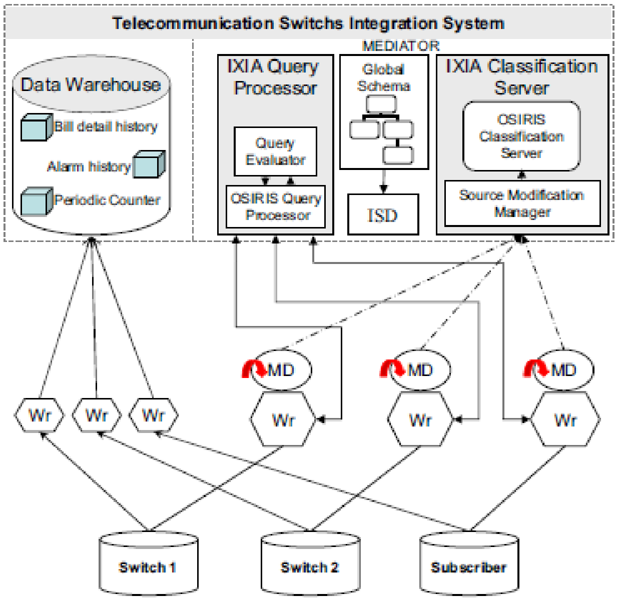

Figure 3: OMC integration system.

Other queries which we want to respond by the integration system, are requested using a mediator system. In this integration scenario, different switches have different delays for sending data from their modules to their databases. In addition, we have not access to their database source and querying them via specific APIs is the only way to access their data. Subscriber database, however, is developed in the telecom company and access to its database is possible. In the other hand, the query response time to extract switch data is crucial for telecom carrier as well as for subscriber that use telecom Web site.

4. Conclusion

A hybrid approach data integration is to develop a partially materialized integrated schema. Fully materialized and fully virtual data integration approaches obey to different priorities. In a fully materialized approach, the main priority is the query response time, and in fully virtual data integration, data freshness is more important.

However, in many data integration scenarios different priorities may be associated with different data, and a trade-off between query response time and data freshness may be preferred to satisfying only one of these two issues.

[1] Qazvin OMC (Operation and Maintenance Center) project was founded by Qazvin Telecommunication Company and Imam-Khomeini International University, 2007-2009.

A flexible approach which permits some data to be materialized and other data to be virtual can satisfy both of these goals. In the existing hybrid approaches the global view is partitioned into materialized and virtual parts. Some objects or relations are chosen to be materialized and others reside in the local sources and will be extracted at query time.

Contrary to other hybrid approaches, IXIA query optimization is not privileged for querying some materialized data but for all the queries of the integration system. In addition, if a query only uses classifying attributes, it obtains a higher level of optimization, thanks to the materialization of classifying attributes in the Modification Detector.

IXIA proposes a data refreshing solution. The modification detector is the core of this solution. It offers two advantages:

1. Data refreshing for different sources, hence for different data can be done in different time periods.
2. The arrangement of refreshing for different data sources can be changed by changing the frequencies of different MDs, which is a decision of the system administrator.

References

[1] L. M. Haas, Beauty and the beast: The theory and practice of information integration, ICDT, 2007, pp. 28–43.

[2] S. Kermanshahani, H. Ahmad, A. Simonet, M. Simonet, A semantic view-based multi-mediator architecture, IKE, 2007, pp. 197–204.

[3] S. Kermanshahani, H. Ahmad, A. Simonet, M. Simonet, A view based architecture for a multi-level semantic mediator, IKE, 2008.

[4] G. V. Solar, A. Doucet, M´ediation de donn´ees : solutions et probl`emes ouverts, Actes des 2`emes Assises nationales du GdR I3, c´epadu`es ´editions Edition, Nancy, France, 2002.

[5] G. Diallo, Une architecture a base d’ontologies pour la gestion unifiee des donnees structurees et non structurees, PhD Thesis, l’Universite Joseph Fourier, France (2006).

[6] A. Y. Levy, A. Rajaraman, J. J. Ordille, Querying heterogeneous information sources using source descriptions, Proceedings of the Twenty second International Conference on Very Large Databases, VLDB Endowment, Saratoga, Calif., Bombay, India, 1996, pp. 251–262.

[7] S. S. Chawathe, H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. D. Ullman, J. Widom, The tsimmis project: Integration of heterogeneous information sources, IPSJ, 1994, pp. 7–18.

[8] E. Jasper, N. Tong, P. McBrien, A. Poulovassilis, View generation and optimisation in the automed data integration framework (2003).

[9] S. Kermanshahani, Semi-materialized framework: a hybrid approach to data integration, CSTST ’08: Proceedings of the 5th international conference on Soft computing as transdisciplinary science and technology, ACM, New York, NY, USA, 2008, pp. 600–606.

[10] C. G. Bassolet, Approches connexionnistes du classement en osiris. vers un classement probabiliste, Phd thesis, Joseph Fourier - Grenoble1, France (1998).

[11] G.Laperrousaz, Etude et mise en place dun environnement pour la conception et linterrogation dun data warehouse osiris, Mmoire dingnieur cnam, p-89, CNAM, Centre de Grenoble, France (Juin 2000).

[12] A. Simonet, M. Simonet, Classement d’objets et evaluation de requˆetes en osiris, BDA, 1996, pp. 273–.525

[13] M. Roger, A. Simonet, M. Simonet, Bringing together description logics and database in an object oriented model, DEXA ’02: Proceedings of the 13th International Conference on Database and Expert Systems Applications, Springer-Verlag, London, UK, 2002, pp. 504–513.

[14] A. Simonet, Types abstraits et bases de donnees: formalisation du concept de partage et analyse statique de contraintes d’integrite, PhD Thesis, Universite Scientique et Medicale de Grenoble, France (1984).

[15] S. Kermanshahani, IXIA (IndeX-based Integration Approach): A Hybrid Approach to Data Integration, PhD Thesis, l’Universite Joseph Fourier, France (2009).

Shokooh KERMANSHAHANI studied Computer Engineering, B.S. in Esfehan University (1995, Iran), M.S. and Ph.D. in Joseph Fourier University (2003, 2009 France). She is currently assistant professor of computer engineering, Imam-Khomeini International University, Qazvin, Iran.

Hamid Reza HAMIDI received B.S. from Sharif University of Technology (1994, Iran), M.S. from Tehran University (1997, Iran) and Ph.D. from Institute National Polytechnic de Grenoble (2005, France). He is currently assistant professor of computer engineering, Imam-Khomeini International University, Qazvin, Iran.