SEMANTIC INTEGRATION OF DIGITAL LIBRARIES*

José Francisco Aldana-Montes, Ismael Navas-Delgado, María del Mar Roldán-García
Computer Languages and Computing Science Department. University of Malaga. Spain.

*This work has been supported by the MCyT grant (TIC 2002-04186-C04-04) convinced that conceptual meditation (as offered in our architectural technical solution) is a very good alternative for integrating DLs.

Keywords: Semantic Mediation, Integrating Heterogeneous Digital Libraries, Ontologies

Abstract: Semantic Integration/Interoperability of heterogeneous data sources offers several key features. These characteristics include the following: publication of the data source's semantics, expressive query capabilities (viz. semantic queries), dynamic integration of data sources, interoperability between semantic integration systems that cooperate in the same or in different application domains, etc. The integration of digital libraries and their interoperability are two of the main issues that still have to be addressed by the Digital Libraries community. In this paper we present the application of a semantic mediator architecture to the domain of Digital Libraries and by means of several use cases we show the main advantages it offers.

1 INTRODUCTION

Digital Libraries (DLs) are vast collections of entities stored and maintained by multiple information sources, including databases, image banks, file systems, etc. Characteristics of DLs include (Adam et al., 2000): (1) massive amounts of data; (2) structured, semi-structured, and unstructured data; (3) frequent modification of the information sources. Therefore, integrating them is a central issue in order to facilitate user access to them. Currently, there is still very little interoperability across heterogeneous digital library systems. Interoperability means the cooperation between these heterogeneous and distributed information sources in a way that is transparent to the user, maintaining their autonomy. Portability, data exchange, scalability, federation, extensibility and open network architectures are also noteworthy research issues in DLs (Borgman, 1999).

Database techniques could be very fruitfully applied to the DLs field. Database research mainly deals with efficient storage and retrieval and with powerful query languages. This community has been seriously disturbed with the massive increment of data sources in the web and the need to integrate them. Large amounts of integration techniques have been developed in the past few years. Furthermore, in recent years, mediators have emerged as a way of integrating databases, using wrappers to translate the mediation architecture at all its points. That is, we hope that by distributing all mediation components we can obtain a more scaleable and reusable different data source information to a common model.

(Ibrahim et al., 2001) and (Adam et al., 2000) are good studies about integration in DLs. In addition to the wrapper-mediator approach, they analyse other approaches that have also been widely studied by the database community; However, mediators are the most common approach for heterogeneous data source integration.

On the other hand, although we believe that mediation is a good approach for integrating digital library information sources, mediators present some deficiencies related to the reusability of common wrapper components, high coupling of wrapper and mediators, and the difficulty of adding new sources dynamically. To provide good integration systems to digital library users, it is necessary to find a solution for these deficiencies. Furthermore, we believe that DLs are more suitable for conceptual, rather than structural, browsing and navigation. In order to solve these problems, we propose a novel mediation architecture which aims at making wrappers independent entities and eliminating their ties to the mediator, thus increasing its reusability in different applications and contexts. It entails additional advantages providing elements to obtain major interoperability among integration systems, that cooperate in the same application domain or have certain relations, such as digital libraries.

Our proposal tries to go beyond traditional application. The focus of this paper is to integrate DLs by using this proposed architecture for semantic mediation (see section 4). We are firmly convinced

Francisco Aldana-Montes J., Navas-Delgado I. and del Mar Roldán-García M. (2004).
SEMANTIC INTEGRATION OF DIGITAL LIBRARIES.
In *Proceedings of the Sixth International Conference on Enterprise Information Systems*, pages 313-318

that conceptual mediation (as offered in our architectural technical solution) is a very good alternative for integrating DLs.

2 RELATED WORKS

The wrapper-mediator approach provides an interface to a group of (semi) structured data sources, combining their local schemas into a global one and integrating the information of local sources. So the views of the data that mediators offer are coherent, performing semantic reconciliation of the common data model representations carried out by the wrappers. Some good examples of wrapper-mediator systems are TSIMMIS (Papakonstantinou et al., 1995) and Manifold (Levy et al., 1996). Several improvements have been made of traditional mediators. One of the most important is the use of standard representation languages, like XML. Thus, MIX (Baru et al., 1999a) (the successor to the TSIMMIS project) and MOCHA (Rodriguez et al., 2000) projects are XML-based.

The next level of abstraction on Web integration corresponds to ontology-based systems. Their main advantage with respect to mediators is their capacity to manage schemas that are unknown a priori. This is achieved by means of a mechanism that allows contents and query capabilities of the data source to be described declaratively. OBSERVER (Mena et al., 1996) uses different ontologies to represent information data sources. Users explicitly select the ontology that will be used for query evaluation. The existence of mappings among ontologies allows the user to change the ontology initially selected.

Model-Based Mediation (Ludascher et al., 2001) is a paradigm for data integration in which data sources can be integrated, taking advantage of an auxiliary expert knowledge. This knowledge includes information about the domain and it is the glue that joins data source schemas together. The expert knowledge is captured in a data structure called Knowledge Map. In Model-Based Mediation the mediation architecture is extended, carrying data sources from the data level without semantics to the conceptual model level. This architecture introduces semantics into data sources and mediators, but they are not published and accessible to agents or applications. Mediators are monolithic systems and they are strongly coupled to wrappers, limiting dynamic integration and interoperability.

In (Navas et al., 2004) we proposed an architecture for semantic mediation. This architecture includes directories in which an ontology and several resources with semantics relevant to the domain information are published. We also improve the wrapper generation process by publishing them as web services and making their semantics accessible. As a result of this evolution from traditional wrappers to data services, the following objectives were achieved: (1) data services can be used by other applications (maybe mediators); (2) semantics of the data services is published on the web and is available for other applications; (3) Wrapper query capabilities can be enveloped into one or more services.

There are some previous works that use traditional mediator architectures in the domain of DLs. In (Baru et al., 1999b), a prototype to integrate DLs using MIX technology is presented. This is a traditional monolithic approach; therefore reusability of mediator components, expressive query capabilities as well as dynamic data source integration are not possible. (Melnik et al.,2000) introduces reusability of mediator components. It proposes an infrastructure to integrate DLs that allows mediators to be composed from a set of modules. However, it does not deal with the other problems mentioned. Section 3 describes how all these problems can be addressed in the context of semantic mediation (Navas et al., 2004).

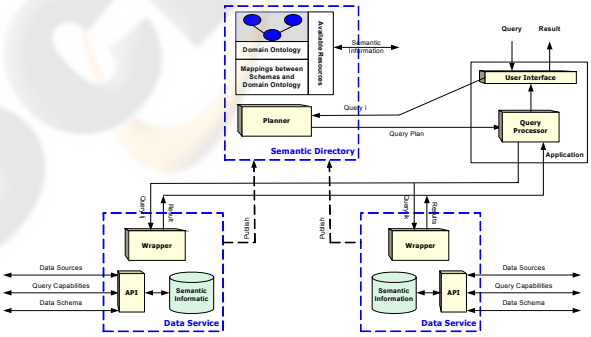


Figure 1: Architecture for Semantic Integration

3 SEMANTIC MEDIATION

Our Semantic Mediation Architecture seeks to make wrappers independent entities and to eliminate their ties to the mediator, thus increasing their reusability in different applications. We emulate P2P (Schollmeier, 2001) hybrid systems, which implement a directory with location information of available resources. In these systems the applications access the resources directly by means of point to point connections provided by the directory.

Our proposal for semantic mediation stems from the need in several domains for dynamic integration, and from two main considerations on the basic architecture of mediation: (1) on the one hand the isolation of wrappers, which are encapsulated as web services (W3C, 2002) (Data Services for us); and (2) on the other hand, the added directory (Semantic Directory) with information about these Data Services (See Figure 1). This architecture allows wrappers to contribute data, schemas of information and query capabilities in a decentralized and easily extensible way. Public interfaces of data services and semantic directories will allow other applications, which share its communication protocol, to take advantage of knowledge about available directory resources. Next we briefly present the components of the proposed architecture.

3.1 Semantic Directory

Semantic directories are at the core of this architecture because they provide essential services for solving user queries. We can define a semantic directory as "a server that offers information about available web resources (data services), a domain ontology, mappings between resource schemas and this ontology, and provides a query planner".

A semantic directory stores an ontology described with OWL (OWL, 2003), which must be generic for the application domain. This ontology describes the core knowledge that is shared by a set of users. Information about data services will be added to a semantic directory when services register in it. This information includes the Resource's Schemas, the location of these resources (the URL of the Data Service, the Query Web Method, etc.) and several mappings between the domain ontology and the resource's schemas. Note that all this information allows the system to solve any kind of query, and not only predefined queries like most mediation systems.

3.2 Data Services

Semantic directories offer essential services for query processing, and data services provide minimal elements for solving queries. We have designed an extensible and adaptive architecture in which we can define a data service as "a service that offers wrapper query capabilities using web protocols". That is, this type of service will solve specific queries for a data source and offer its query capabilities as a web service. The publication of these online web services using UDDI (Universal Description Discovery Integration) (UDDI, 2003) could allow other applications to dynamically discover wrappers by means of an easy interface. However, our data services have been devised for publication in a specialized and previously described type of directory: the semantic directory. Thus, a data service needs to be registered in one or more semantic directories in order to be used by a mediator or other software agent.

4 USE CASES

In this section we present a use case of the proposal described for the integration of digital library data sources. After that, we describe several advanced queries that highlight the advantages of the proposed architecture. As a first step we have generated a domain ontology to be used in a semantic directory. This ontology represents the domain knowledge of a group of digital library users (see Figure 2). This ontology is based on terms described by the Dublin Core Metadata Initiative (Dublin, 2003). Then we implement the application that will use this directory and includes an evaluator and user interface.

Once semantic directories have been developed, they are autonomous and do not need human actions, but a semantic directory and an application are not enough to solve user queries. In order to illustrate how our architecture works we present a simple example, together with all the elements that are necessary to solve the query example. Suppose that we have developed several data services about computer science publications and added them to the semantic directory. For example, we can add to our system data services that access to the BNE (Spanish National Library), DBLP (Database & Logic Programming) and CSB (The Collection of Computer Science Bibliographies).

Now, we can solve a query like: "Find articles whose author is Ullman and were published the same year as the book titled 'Data on the Web'".

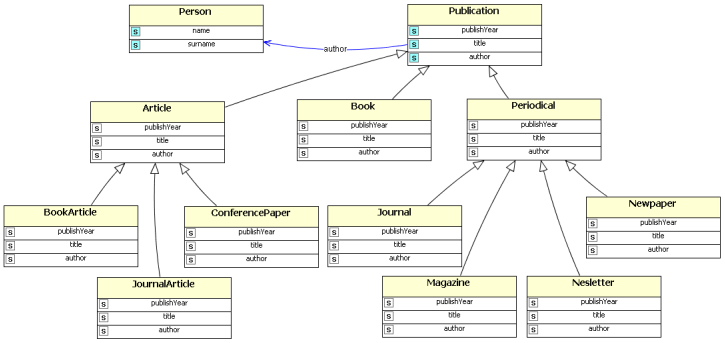


Figure 2: Domain Ontology

This query is represented internally by the user interface in logical terms:

ans(A) :- Article(A) , author(A,P), surname(P,"Ullman") , publishYear(A,Y) ,
Book(B) , publishYear(B,Y) , title(B,"Data on the Web")

This query is sent to the semantic directory, which analyzes it taking advantage of mappings between resources and the domain ontology (Figure 3a shows an example of mapping between part of the CSB schema and part of the domain ontology). In our example we map the data retuned by the BNE to the Book class, the data retuned by the DBLP to Article and Book classes and the data returned by the CSB to the ConferencePaper, JournalArticle, and Book classes.

Taking into account all these mappings, the proposed query is divided into two sub-queries:

(1) ans(Y) :- Book(B) , title(B,"Data on the Web") , publishYear(B,Y)
(2) Q(Y): ans(A) :- Article(A) , author(A,P), name(P,"Ullman") , publishYear(A,Y)

Note that the second sub-query needs the year of publication from the first sub-query, so the evaluation must be sequential. Now, these queries must be evaluated in resources in which they can be solved. For this task the semantic directory also makes use of mappings, and determines that the first query can be solved in all resources. Thus, it is sent to the resources, and we obtain inconsistent results (see Figure 3b), which are resolved in this case with the result that most resources return. However, we could use quality measurements of each resource in order to solve possible inconsistencies, and apply a balanced average.

Once the year of publication (1999) has been obtained, we can try to solve the second sub-query, and finally achieve the user-query result. If we perform this task with a traditional mediator (that uses an ontology as the integration schema), we can only send this query to the DBLP data service. However, using our architecture, before solving queries we can use inference mechanisms to get better query plans. Thus, we use the class-subclass inference mechanism to obtain that a ConferencePaper, a BookArticle and a JournalArticle are also instances of Article class (Figure 2). Using this knowledge our architecture sends the second sub-query to the DBLP and CSB data services (Figure 4). In this way, we can obtain more results than if we only use the first service, which is what happens in traditional mediators. Finally, sub-query results are composed in order to obtain ontology instances that will be returned to the end-user. In our example these instances include received data, but the system removes duplicate instances of the paper titled "Optimizing Large Join Queries in Mediation Systems". Note that the instance of the paper "Computing capabilities of mediators" will not be returned by the traditional mediator, because we have obtained them taking advantage of the class-subclass inference capabilities of our system.

4.1 Advanced Queries

This section tries to illustrate differences

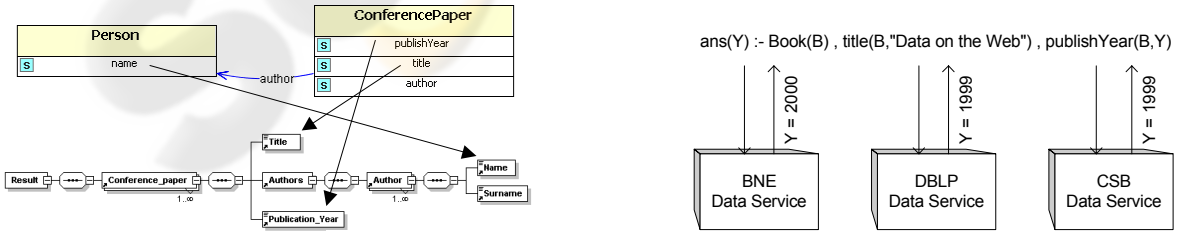
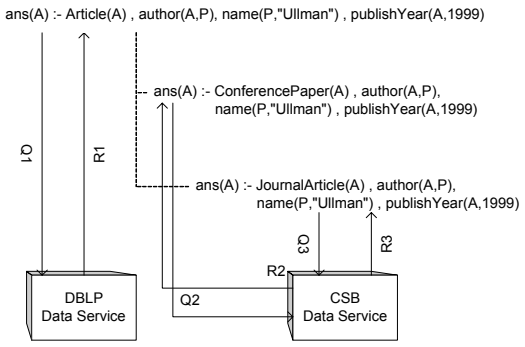


Figure 3: (a) Mapping example; (b) First sub-query evaluation



R1	Article: Title	Authors	Year
	Optimizing Large Join Queries in Mediation Systems	Ramana Yerneni Jeffrey Ullman	1999
	The MIDAS Data-Mining Project at Stanford	Jeffrey Ullman	1999
	Some Advances in Data-Mining Techniques	Jeffrey Ullman	1999
	Answering Queries Using Limited External Query Processors	Alon Levy Anand Rajaraman Jeffrey Ullman	1999

R2	ConferencePaper: Title	Authors	Year
	Optimizing Large Join Queries in Mediation Systems	Ramana Yerneni Jeffrey Ullman	1999
	Computing capabilities of mediators	Jeffrey Ullman	1999

R3	JournalArticle: Title	Authors	Year

Figure 4: Second sub-query evaluation

between traditional mediators and the proposed architecture with respect to query complexity. By exploiting the reasoning capabilities of ontologies it is possible to solve queries that are impossible to solve by means of traditional systems. These reasoning capabilities allow us to derive new knowledge that the ontology does not describe explicitly. Ontology definition languages like OWL provide the possibility of adding rich semantic information to the ontologies; this information can be used to infer knowledge that helps us improve our queries.

In order to clarify these ideas we present some extensions of the previous ontology and we describe how we can use the ontology knowledge to derive new knowledge. Suppose that we add to the ontology some knowledge about which classes are disjoint classes. For example, an article and a technical report cannot be a thesis, thus Article and Thesis are disjoint classes and TechnicalReport and Thesis too (Figure 5a). Since a publication has to be an article, a book or a technical report and always one of them, we can infer that every thesis is also a book. Therefore, a query such as "Find all books whose author is Ullman" will return all books written by Ullman plus his thesis (because we infer that a thesis is also a book). A traditional mediator has no mechanism to identify this information, so only Ullman's books would be returned.

The next example shows that it is possible to infer that a class has no instances and also that two classes are equivalent (contain the same set of instances). For example (Figure 5b), if we know that an article has to be a Journal article or a book and always one of them, and every book is also a Master Thesis, we can infer that each article is also a journal article and therefore, that no articles are book articles (i.e. BookArticles class has no instances).

Finally, the ontology definition language allows us to define special properties of relationships. These properties also encapsulate knowledge about the domain that it is not explicitly asserted. For example, we can define a relationship as a transitive one. Figure 6 shows an extension of our ontology where the relationship Contains (defined between two Publications) is a transitive relationship. Therefore, we could assert that a Journal contains Journal Papers and a Journal Paper contains abstracts, because Journal, JournalPaper and Abstract are subclasses of Publication. Since the relationship "contain" is transitive, we can infer that a Journal contains Abstracts. This information can also be used to make a query like "find all Abstracts contained in the Journal title Journal of Digital Libraries". In a traditional mediator it is not possible to know that a Journal has Abstracts, because this information would not be in the integration schema.

5 CONCLUSIONS AND FUTURE WORK

In this paper we present an architecture to integrate Digital Libraries which is based on an extension of traditional mediation, called semantic mediation. A mediator typically uses an integration schema as a global view of the local schemas of the data sources that are integrated. Thus, queries are limited to the information that the integration schema provides. In our approach, the semantics introduced in the Semantic Directories allows users to make more expressive queries, viz. semantic queries.

Furthermore, information inferred from the ontology-explicit knowledge is used to make queries that a traditional mediator could not evaluate. Therefore, in several domains in which there are no technical users, such as librarians, dynamic integration is a very important issue.

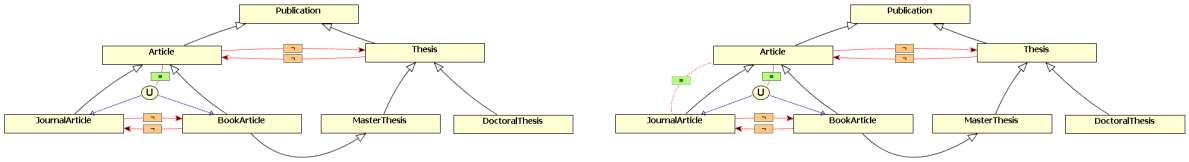


Figure 5: Extending the ontology with knowledge about (a) disjoint classes (b) equivalent classes

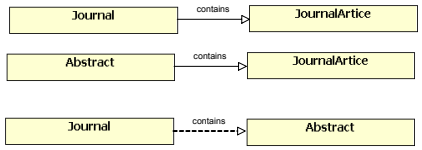


Figure 6: Extending the ontology with knowledge about transitive relationships

In this context, it is necessary to give users a simple environment for integrating data information without modifying the mediator code. The directories supply an easy way to integrate data sources, opening up new directions for dynamic integration.

As future work, we intend to study the possibility of giving data services more semantics, taking into account service quality, relations with other domain ontologies, etc. Besides, the scalability of this architecture will provide the possibility of integrating not only Digital Library services but also semantic directories, making possible a full semantic integration of resources and the interoperability between applications and between different domains. This will allow us to make complex queries, relating domains like Digital Libraries and Molecular Biology. These kinds of queries will really exploit the advantages of the proposed architecture. For example, we could query a Digital Library using both the results of a biological experiment stored in a Biological data source and the related specific domain knowledge.

REFERENCES

Adam, N. R., Atluri, V., Adiwijaya, I.. 2000. *SI in Digital Libraries. Communications of the ACM*, vol. 43, no. 6, pages 64-72.

Baru, C., Gupta, A., Ludäscher, B., Marciano, R., Papakonstantinou, Y., Velikhov, P., Chu, V.. 1999. XML-based Information Mediation with MIX. *In Demostrations, ACM/SIGMOD,* pages 597-599.

Borgman, C.L.. 1999. What are digital libraries? *Competing visions. Information Processing and Management* 35, 227-243.

C. Baru, A. Gupta, V. Chu, B. Ludaescher, R. Marciano, Y. Papakonstantinou, and P. Velikhov. 1999. XML-Based Information Mediation for Digital Libraries. In Demo Session, ACM Digital Libraries'99.

Dublin Core Metadata Initiative. http://dublincore.org/

Ibrahim, I. K., Schwinger, W.. 2001. Data Integration in Digital Libraries: Approaches and Challenges, *Proceedings of the International Seminar on Digital Library and Knowledge Management*.

Levy, A., Rajaraman, A., Ordille, J.. 1996. Querying Heterogeneous Information Sources Using Source Descriptions. *In Proc. VLDB,* pages 251-262.

Ludascher, B., Gupta, A., Martone, M. E.. 2001. *Model-based Mediation with Domain Maps.* In ICDE'01. pp. 81-90.

Melnik, S., Garcia-Molina, H., Paepcke, A.. 2000. "A Mediation Infrastructure for Digital Library Services," *Proc. ACM Digital Libraries Conf*., ACM Press.

Mena, E., Kashyap, V., Sheth, A., Illarramendi, A.. 1996. OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies. *Conference on Cooperative Information Systems.*

Navas, I., Aldana, J.F.. 2004. Towads Conceptual Mediation. ICEIS.

OWL Web Ontology Language 1.0 Reference. 2003. http://www.w3.org/TR/owl-ref.

Papakonstantinou, Y., Garcia-Molina, H., Widom, J.. 1995. Object Exchange Across Heterogeneous Information Sources. In Proc. ICDE.

Rodriguez-Martinez & Roussopoulos. 2000. “MOCHA: Self-Extensible Database Middleware System for Distributed Data Sources”. *In Proc. of the ACM SIGMOD Conference*.

Schollmeier, R.. 2001. A Definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications. *Peer-to-Peer Computing:* 101-102.

UDDI Spec Technical Committee Specification. http://uddi.org/pubs/uddi-v3.0.1-20031014.pdf.

W3C Web Services Activity. http://www.w3.org/2002/ws/