

Demonstration of the COntext INterchange Mediator Prototype*

S. Bressan, C.H. Goh[†], K. Fynn, M. Jakobisiak, K. Hussein,
T. Lee, S. Madnick, T. Pena, J. Qu, A. Shum, M. Siegel

Sloan School of Management, Massachusetts Institute of Technology

Email: context@mit.edu

Abstract

The *Context Interchange* strategy presents a novel perspective for mediated data access in which semantic conflicts among heterogeneous systems are not identified a priori, but are detected and reconciled by a *Context Mediator* through comparison of *contexts*.

1 COIN

The *Context Interchange* (COIN) project aims to develop tools and technologies for supporting access to heterogeneous and distributed information systems.

The underlying integration strategy [GBMS96, SSR94], called the COIN strategy, presents a novel perspective for mediated data access in which semantic conflicts among heterogeneous systems are not identified a priori, but are detected and reconciled by a *Context Mediator* [Wie92] through comparison of *contexts* (i.e. the *contexts* associated with the sources and receivers engaged in data exchange).

The COIN *framework* [GBMS96] is a mathematical structure offering a sound and formal foundation for this strategy and for guiding its implementation. The COIN *framework* comprises a *data model* and a logical *language* of the family of the *Frame-Logic* [KL89]. The data model and the language are used to define the *domain model* of the application and the *contexts*. The data model contains the definitions of the semantic-types which constitute the domain of discourse. The *contexts*, associated with both information *sources* and *receivers*, are collections of statements defining *how* data should be interpreted and *how* potential *conflicts* (differences in the interpretation) may be resolved. Concepts such as *semantic-objects* and *semantic-comparisons*, *modifiers*, and *conversion functions* allow the semantics of data to be defined both within and across the *contexts*. They have been made explicit and built-in the COIN data model and language. Together with the deductive and object-oriented

features inherited from the Frame-Logic, they constitute an appropriate and expressive support for both the representation of semantic knowledge and the reasoning about semantic heterogeneity.

Context mediation is the process of rewriting queries posed in the receiver's *context* into a set of *mediated* queries where all potential conflicts are explicitly resolved. This process is based on an *abductive* procedure [KK93] which determines, according to the statements in the different *contexts* involved, what information is needed to answer the query, and what and how conflicts may be resolved.

From a system perspective, the COIN strategy combines the best features of *loose-* and *tight-coupling* approaches to *semantic interoperability* among autonomous and heterogeneous systems by allowing the complexity of the system to be harnessed in small chunks, by enabling sources and receivers to remain loosely-coupled to one another, and by sustaining an infrastructure for data integration. The integration approach is not only *non-intrusive* but also *scalable*, *extensible* and *accessible* [GMS94]. By *scalability*, we require that the complexity of creating and administering (maintaining) the interoperation services should not increase exponentially with the number of participating sources and receivers. *Extensibility* refers to the ability to incorporate changes in a graceful manner; in particular, local changes should not have adverse effects on other parts of the larger system. Finally, *accessibility* refers to how the system is perceived by a user in terms of its ease-of-use and flexibility in supporting different kinds of queries.

2 The Prototype

Although, the strategy applies to a variety of application scenarios, the COIN project focuses on the integration of databases and semi-structured information sources distributed over public or corporate Internet based information infrastructures. Users and application programs (e.g. users of Web-browsers, spreadsheets, data-warehouses) have transparent access to the remote information sources (e.g. on-line databases, web-sites) through a server providing the mediation services.

Figure 1 illustrates the architecture of the context mediation infrastructure deployed in an application.

The sources we consider are ranging from on-line databases (e.g. an Oracle database on our picture) to semi-structured Web-sites. In order to provide a uniform access to this sources, a set of components called *wrappers* are used as interfaces between the mediation engines and the sources. The wrappers are not only communication gateways between

*This work is supported in part by DARPA and USAF/Rome Laboratory under contract F30602-93-C-0160.

[†]Financial support from the National University of Singapore is gratefully acknowledged.

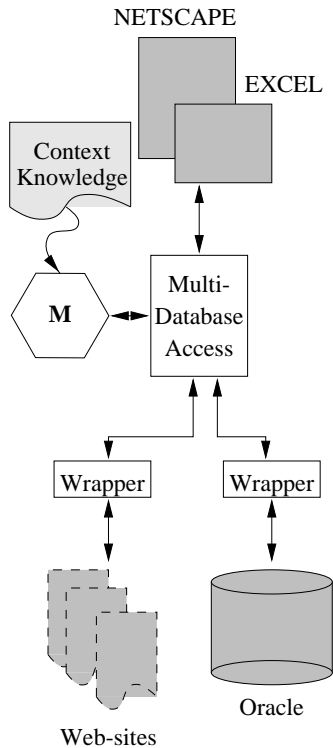


Figure 1: Architectural overview of the Context Interchange Prototype.

the multi-database access engine and the sources, but they also provide a SQL interface to any source including the Web-sites and deliver answers to the queries in a relational table format. The Web wrapping technology we have developed [Qu96] is based on a high level declarative language for the specification of what information can be extracted. A program in the specification language defines a transition network corresponding to the possible transitions from one Web-page to another, and regular expressions corresponding to what information is located on a page.

On the receiver's side we propose an Application Programming Interface (API) of the family of the Object Database Connectivity protocol (ODBC). The protocol supporting this API is currently tunneled in the HyperText Transfer Protocol (HTTP) of the World Wide Web. The API can be used within any application with basic capabilities for Internet socket based communication. However, we have developed two types of ready-to-use interfaces: A HyperText Markup Language (HTML) Query-By-Example (QBE) and an ODBC driver which gives access to the mediation services to any Windows95 and WindowsNT ODBC compliant applications such as Microsoft Excel or Microsoft Access.

The Multi-database access engine constitutes a front-end of dictionary and query services to the multiple wrapped sources. Its main functions are:

- Serving schema information such as names and attribute types of the table located in the various sources;
- Planning and optimizing the multi-source queries taking the sources capabilities, and the execution and communication costs into account;

- Controlling the execution of the resulting query execution plan and executing the necessary local operations (e.g. joins across sources).

For the management of dictionary information and in order to handle large results or large sets of temporary data, the multi-database access engine uses two local secondary storages.

The mediation engine intercepts a query to the multi-database engine and rewrite it according to the context knowledge it has about the receiver and the sources involved. The rewritten query is usually a union of sub-queries corresponding respectively to the possible conflicts between the context assumptions and their resolution.

3 A short Example of Mediation

Let us consider, for instance, the query "What are the names and revenues, of the companies whose revenue is bigger than their expenses?". Let us assume that such a query involves two sources and one relation in each source. The tables in the sources and the ancillary web source reporting currency exchange rates are shown on figure 2. The query is expressed in SQL:

```
SELECT r1.cname, r1.revenue FROM r1, r2
WHERE r1.cname = r2.cname
AND r1.revenue > r2.expenses;
```

R1

IBM	100 000 000	USD
NTT	100 000 000	JPY

R2

IBM	1 500 000
NTT	5 000 000

WWW	
USD	JPY
104.00	

Figure 2: The relations R1 and R2, and the currency exchange Web source.

The above query, however, does not take into account the fact that data sources are administered independently and have different *contexts*: i.e., they may embody different assumptions on how information contained therein should be interpreted. For instance, the data reported in the two sources differ in the currencies and scale-factors of company financials (i.e., financial figures pertaining to the companies, which include revenue and expenses). Specifically, in Source 1, all company financials are reported using the currency shown and a scale-factor of 1; the only exception is when they are reported in Japanese Yen (JPY) in which

case the scale-factor is 1000. Source 2 reports all company financials in USD using a scale-factor of 1. In the light of these remarks, the (empty) answer returned by executing Q1 is clearly not a “correct” answer since the revenue of NTT (9,600,000 USD = 1,000,000 × 1,000 × 0.0096) is numerically larger than the expenses (5,000,000) reported in r2. The query is rewritten by the mediation engine into:

```
SELECT r1.cname, r1.revenue
FROM r1, r2
WHERE r1.currency = 'USD'
AND r1.cname = r2.cname
AND r1.revenue > r2.expenses;
UNION
SELECT r1.cname, r1.revenue * 1000 * r3.rate
FROM r1, r2, r3
WHERE r1.currency = 'JPY'
AND r1.cname = r2.cname
AND r3.fromCur = r1.currency
AND r3.toCur = 'USD'
AND r1.revenue * 1000 * r3.rate > r2.expenses
UNION
SELECT r1.cname, r1.revenue * r3.rate
FROM r1, r2, r3
WHERE r1.currency <> 'USD'
AND r1.currency <> 'JPY'
AND r3.fromCur = r1.currency
AND r3.toCur = 'USD'
AND r1.cname = r2.cname
AND r1.revenue * r3.rate > r2.expenses;
```

The mediated query considers all potential conflicts between relations r1 and r2 when comparing values of “revenue” and “expenses” as reported in the two different *contexts*. Moreover, the answers returned may be further transformed so that they conform to the *context* of the receiver. Thus in our example, the revenue of NTT will be reported as 9 600 000 as opposed to 1 000 000. More specifically, the three-part query shown above can be understood as follows. The first sub-query takes care of tuples for which revenue is reported in USD using scale-factor 1; in this case, there is no conflict. The second sub-query handles tuples for which revenue is reported in JPY, implying a scale-factor of 1000. Finally, the last sub-query considers the case where the currency is neither JPY nor USD, in which case only currency conversion is needed. Conversion among different currencies is aided by the ancillary data source r3 (a Web service) which provides currency conversion rates. This second query, when executed, returns the “correct” answer consisting only of the tuple <‘NTT’ 9 600 000>.

4 conclusion

Together with our industrial partners, we are currently deploying our technology in several experimental applications in particular in the area of financial analysis decision support (profit and loss analysis, marketing intelligence). We have build demonstrations accessing several financial and company profile on-line databases and using Web sites as primary (for instance, sites reporting every 15 minutes security prices in the various stock exchanges) or ancillary (for instance, date conversion services or currency exchange) information sources.

References

- [GBMS96] C. H. Goh, S. Bressan, S. E. Madnick, and M. Siegel. Context Interchange: Representing and Reasoning about Data Semantics in Heterogeneous Systems. Technical Report #3928, Sloan School of Management, MIT, 50 Memorial Drive, Cambridge MA 02139, October 1996.
- [GMS94] C. H. Goh, S. Madnick, and M. Siegel. Context interchange: overcoming the challenges of large-scale interoperable database systems in a dynamic environment. In *Proceedings of the Third International Conference on Information and Knowledge Management*, pages 337–346, Gaithersburg, MD, Nov 29–Dec 1 1994.
- [KK93] A. Kakas and F Kowalski, R and. Toni. Abductive logic programming. *Journal of Logic and Computation*, 2(6):719–770, 1993.
- [KL89] M. Kifer, , and G. Lausen. F-Logic: a higher-order language for reasoning about objects, inheritance and scheme. In *Proc ACM SIGMOD*, pages 134–146, 1989.
- [Qu96] J. Qu. Data wrapping on the world wide web. Technical Report CISL WP#96-05, Sloan School of Management, Massachusetts Institute of Technology, February 1996.
- [SSR94] E. Sciore, M. Siegel, and A. Rosenthal. Using semantic values to facilitate interoperability among heterogeneous information systems. *ACM Transactions on Database Systems*, 19(2):254–290, June 1994.
- [Wie92] G. Wiederhold. Mediators in the architecture of future information systems. *IEEE Computer*, 25(3):38–49, March 1992.