Metadata Jones and the Tower of Babel:
The Challenge of Large-Scale Semantic Heterogeneity

Stuart E. Madnick
Sloan School of Management, Massachusetts Institute of Technology
30 Wadsworth Street, Room E53-321
Cambridge, MA USA 02139
Phone: (617) 253-6671
Fax: (617) 253-3321
smadnick@mit.edu

ABSTRACT

The popularity and growth of the "Information SuperHighway" (e.g., the Web) have dramatically increased the number of information sources available for use and the opportunity for important new information-intensive applications (e.g., massive data warehouses, integrated supply chain management, global risk management, in-transit visibility). Unfortunately, there are significant challenges to be overcome regarding *data extraction* and *data interpretation* in order for this opportunity to be realized.

Data Extraction: One problem is the difficulty in easily and automatically extracting very specific data elements from Web sites for use by operational systems. New technologies, such as XML and Web Querying/Wrapping, offer possible solutions to this problem.

Data Interpretation: Another serious problem is the existence of heterogeneous contexts, whereby each SOURCE of information and potential RECEIVER of that information may operate with a different context, leading to large-scale semantic heterogeneity. A context is the collection of implicit assumptions about the context definition (i.e., meaning) and context characteristics (i.e., quality) of the information. As a simple example, whereas most US universities grade on a 4.0 scale, MIT uses a 5.0 scale – posing a problem if one is comparing student GPA's. Another typical example might be the extraction of price information from the Web: but is the price in Dollars or Yen (If dollars, is it US dollars or Hong Kong dollars), does it include taxes, does it include shipping, etc. – and does that match the receiver's assumptions?

In this paper, examples of important context challenges will be presented and the critical role of metadata, in the form of context knowledge, will be discussed.

Preamble

The Bible tells the tale of the Tower of Babel where mankind endeavored to build a tower to reach to the Heavens. According to the Bible, God introduced a multiplicity of languages – the resulting confusion made it impossible for such large-scale coordination and communication and led to the termination of the tower's construction. Today we are attempting to build "information superhighways" to access information from around the organization and around the world. Will this current great endeavor succeed or will it also be overcome by a "confusion of tongues"? The effective use of metadata can provide an approach to overcoming the challenges.

Motivation

There have been significant research efforts focused on physical information infrastructure, such as establishing high-speed data links to access information distributed throughout the world. It is increasingly obvious, however, that this kind of "physical connectivity" alone is not sufficient since the exchange of bits and bytes is only valuable when information can be efficiently and meaningfully exchanged. These capabilities are essential to providing the "logical connectivity" that is critically needed for dealing with the challenges of the information age.

The need for intelligent information integration is important to all information-intensive endeavors, with broad relevancy for global applications, such as Manufacturing (e.g., Integrated Supply Chain Management), Transportation/Logistics (e.g., In-Transit Visibility), Government / Military (e.g., Total Asset Visibility), Financial Services (e.g., Global Risk Management).

I. Distributed Context Knowledge to Integrate Heterogeneous Sources and Uses

Advances in computing and networking technologies now allow huge amounts of data to be gathered and shared on an unprecedented scale. Unfortunately, these new-found capabilities by themselves are only marginally useful if the information cannot be easily **extracted** and **gathered** from **disparate sources**, if the information is represented differently with **different interpretations**, and if it must satisfy **differing user needs**.

Some of the extraction and dissemination challenges arise because the information sources may be traditional databases, web sites, or even spreadsheets or electronic mail. Furthermore, the user may originate his or her request in a variety ways. Even more challenging to the correct interpretation of information is the fact that the sources and users may each assume different semantics or "context" (as a trivial example, one source may be assuming measurements in meters whereas another assumes feet.)

Contextual issues can be much more complex in other situations. For example, the meaning of "net sales" may vary – with "excise taxes" included for government reporting purposes in one context, but excluded for security analysis purposes in another. Also, one context may use information for a fiscal year as reported by the company, while another may use a standardized fiscal year to make all companies comparable. Furthermore, there may be multiple users that might want an answer to such a question, each with their own desired media and meaning (user context profile). Note that a "user" might be a person, an application program, a database, or a data warehouse.

In summary, to exploit the proliferation of information sources that are becoming available, we require not only technology, such as the Internet, that will provide "physical connectivity" to information sources, but also "logical connectivity" so that the information can be obtained from disparate sources and can be meaningfully assimilated. This **context knowledge** is often widely **distributed** within and across organizations. Solutions adopted to achieve interoperability must be scaleable and extensible. Thus, it is important to support **the acquisition, organization, and effective intelligent usage of distributed context knowledge**. Components of a Context Interchange System have been designed and implemented as a basic prototype at MIT.

II. The Intelligent Information Integration Challenge

Simple Example

As an illustration of the problems created by the disparities underlying the way information is provided, represented, interpreted, and used, consider the example depicted in Figure 1 below. The users wish to answer a fairly common, but important, type of question:

"**How much funds are left for project A**?" The calculation in this case is conceptually quite simple, merely subtract the expenses incurred by the 3 regions from the amount of funds that had been allocated (these are all shown on the left side under the heading labeled "Sources").

> *Although we only discuss this particular example, the reader is encouraged to consider the many other similar situations that exist in all disciplines and among all organizations.*

Information Extraction and Dissemination Challenges

Extraction: Even assuming that all the necessary information is available electronically and connected via the Internet, they may be in differing media and meaning. In this example, the allocated funds are in an Oracle **relational database** in Singapore, the expenses for Region 1 (USA) are available from a **web site**, the expenses for Region 2 (UK) are in an Excel **spreadsheet**, and the expenses from Region 3 (Japan) are provided via a semi-structured **electronic mail** message. In order to compute the desired answer, the information must be extracted from these varying sources and gathered together.

Dissemination: Similarly, the actual request may originate in many ways (these are shown on the right side under the heading labeled "End-User Environments & Applications"). A user in the USA may be making this request from a **Web browser**, a user in the UK may have this request originating from an "embedded SQL query" in a **spreadsheet**, a user in Singapore may be collecting this information for **data warehousing** purposes. Furthermore, this information may be requested and used as part of calculations for arbitrary **application programs** (e.g., preparation of budgeting reports, generation of exception reports, etc.)

Information Interpretation Challenges

Merely subtracting the numbers shown in the Figure 1 expense sources on the left from the allocated number does not produce the "right" answer because different sets of assumptions underlie the representation of the information in the sources. These assumptions are often not explicit, we call these the meaning or *context* of the information. In this case, the source contexts are indicated at the far left in Figure 1.

For the example shown in Figure 1, the allocated funds are expressed in 1000's of Singapore dollars, the expenses in Region 2 are expressed in 1's of British pounds excluding the 10% VAT charges, and Region 3 reports its expenses in 100's of Japanese Yen.

Likewise, the receivers' may have their own unique context, shown at the far right in Figure 1. A USA user may expect the answer in 1's of US dollars, whereas the Singapore user may wish the answer in 1000's of Singapore dollars. The UK user may want the answer in 100's of British pounds including the 10% VAT charges. Under these circumstances, answering even the "simple" question of Figure 1 is not so simple – try it yourself. If fact, auxiliary information sources may be needed, such as currency conversion rates, as well as rules on how such conversions should be done (e.g., as of what date).

Contextual issues can be much more complex in other situations. For example, the meaning of "net sales" may vary – with "excise taxes" included for government reporting purposes in one context, but excluded for security analysis purposes in another. Also, one context may use information for a fiscal year as reported by the company, while another may use a standardized fiscal year to make all companies comparable. Furthermore, there may be multiple users (see right side of Figure 1) that might want an answer to such a question, each with their own desired media and meaning (user context profile). Note that a "user" might be a person, an application program, a database, or a data warehouse.

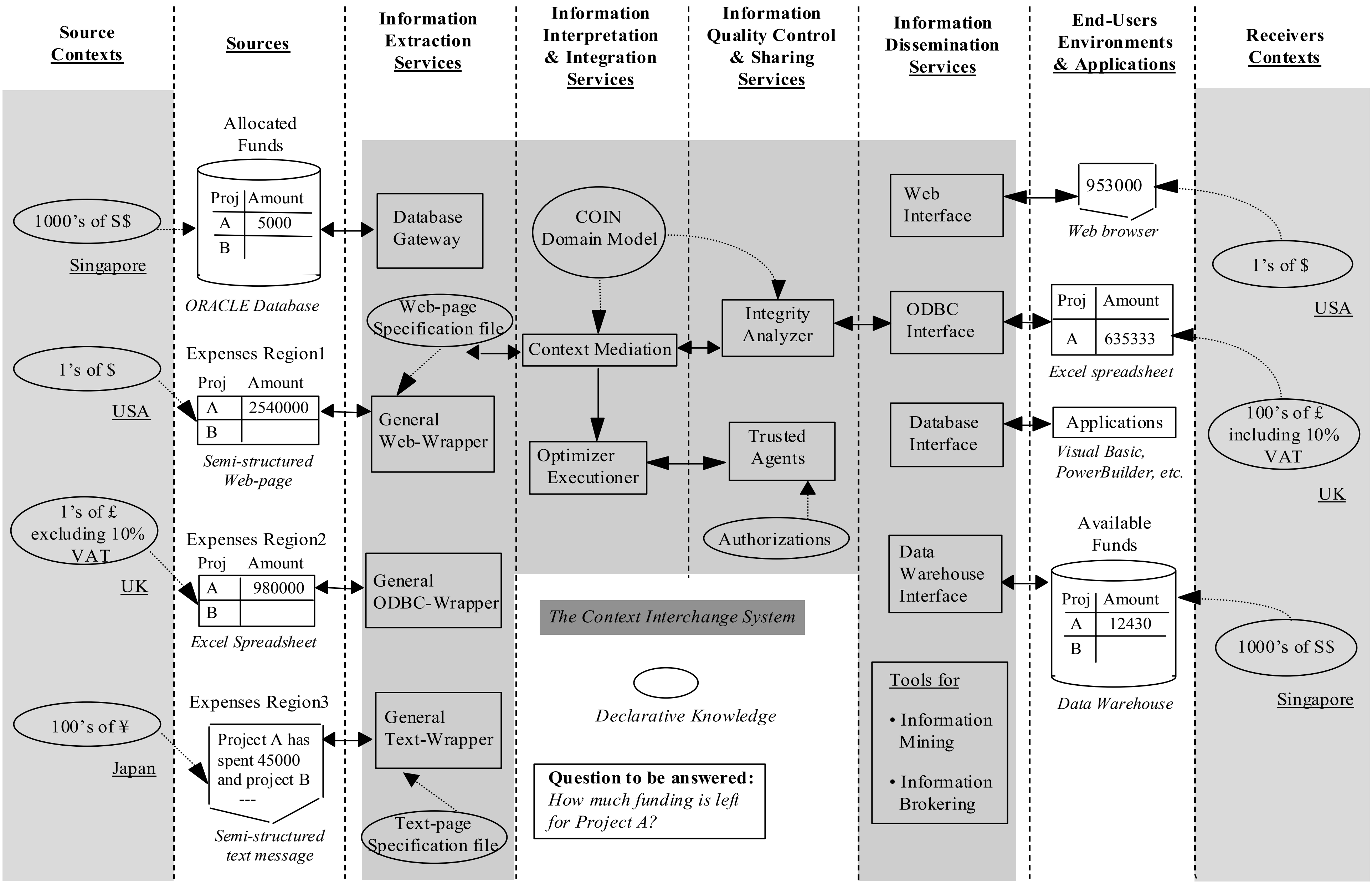

Figure 1. Example Application Illustrating Modular Architecture

In summary, it is increasingly apparent that to exploit the proliferation of information sources that are becoming available, we require not only technology, such as the Internet, that will provide "physical connectivity" to information sources, but also "logical connectivity" so that the information can be obtained from disparate sources and can be meaningfully assimilated. With the amount and diversity of information sources available it is necessary to be able to extract and organize the information from not only structured databases but also semi-structured web sources, spreadsheets, and text sources. In addition solutions adopted to achieve interoperability must be scaleable and extensible and provide decision makers with the appropriate services in an efficient and timely manner in their environments and their applications.

Basic components of a Context Interchange System, illustrated in the center portions of Figure 1, have been designed and implemented as a limited prototype. In one sample application, it makes use of several online databases (e.g., Disclosure, Worldscope, Datastream – historical financial information sources), various web sites (e.g., Security APL – current stock exchange prices, Edgar – USA SEC filings, and Olsen – currency conversion information), and semi-structured documents (e.g., Merrill Lynch analyst reports). The financial information needed to answer a question are extracted from these sources, correctly interpreted (involving automatic conversions), integrated and disseminated in various ways, such as into an Excel spreadsheet application of a financial analyst.

III. Overview of the Context Interchange Approach

1. Context Interchange Architecture.

Context Interchange is a mediation approach for semantic integration of disparate (heterogeneous and distributed) information sources. It has been described in [GBMS96a]. The Context Interchange approach includes not only the mediation infrastructure and services, but also wrapping technology and middleware services for accessing the source information and facilitating the integration of the mediated results into end-users applications.

The architecture comprises three categories of components: the wrappers, the mediation services, and the middleware, interface, and facilitation services.

The wrappers are physical and logical gateways providing a uniform access to the disparate sources over the network.

The set of Context Mediation Services, comprises a Context Mediator, a Query Optimizer and a Query Executioner. The Context Mediator is in charge of the identification and resolution of potential semantic conflicts induced by a query. This automatic detection and reconciliation of conflicts present in different information sources is made possible by general knowledge of the underlying application domain, as well as informational content and implicit assumptions associated to the receivers and sources. These bodies of declarative knowledge are represented in the form of a domain model, a set of elevation axioms, and a set of context theories respectively.

The result of the mediation is a mediated query. To retrieve the data from the disparate information sources, the mediated query is then transformed into a query execution plan, which is optimized, taking into account the topology of the network of sources and their capabilities. The plan is then executed to retrieve the data from the various sources, results are composed as a message, and sent to the receiver.

The middleware, interface, and facilitation services are the services which give access to the mediation services for users and application programs. They rely on an Application Programming interface and a protocol implemented as a standard subset of the Open Data Base Connectivity (ODBC) protocol tunneled into the HyperText Transfer Protocol (HTTP). Examples of interfaces and facilitation services are the Query-By-Example Web interface which is a *point-and-click* interface for the construction of ad-hoc queries [Jako96], and the Context ODBC driver [Shum96] which gives access to the mediation infrastructure to any ODBC-compliant Windows 95 or Windows NT application (Excel, Access, etc.).

2. Wrapping.

Wrappers serve as gateways to external information sources for mediation services engines. While information sources vary widely in interface technology and physical data representation, the wrappers should provide a uniform interface to the sources. Two general classes of information sources are: structured data sources, such as traditional relational DBMS's (Oracle and Ingres), and on-line information services, such as Web sites reached though navigable HyperText Markup Language (HTML) pages.

COIN wrappers [Qu96] present a common client interface with the appearance of a relational table to the mediation services engine. The protocol used at the wrapper interface is identical to the protocol for accessing mediation services – ODBC tunneled into HyperText Transfer Protocol (HTTP). Requests are presented in SQL. Results are returned in the form of standard objects, such as HTML tables or JavaScript objects. Because of the common interface at each stage, a user can, in fact, by-pass mediation services and directly access raw data from a source through a wrapper.

COIN wrappers for relational DBMS's serve as protocol converters. Queries or other access requests are received from the client in COIN protocol. The SQL is extracted and presented to the DBMS using its own API. Query results are then obtained from the DBMS API and delivered to the client using the COIN protocol via HTTP.

For the Web sources, we have developed a generic Web-wrapping technology, which is capable of extracting semi-structured information from Web-services. The COIN Web-wrapping technology is unique for it takes advantage of the Hypertext structure of Web-sources and of the underlying structure provided by the HTML. We treat a Web service as a collection of static and dynamic pages connected by transitions.

Information on the Web is often not contained on a single page, but is distributed over a group of pages linked by static (e.g. <A HREF=**…**>) and dynamic hypertext links (e.g. <FORM ACTION=**...**>). In fact, whether a "service" is located on a single Web-server, or distributed over a number of independently maintained sites, is transparent to the user. Typically, a user may contact the "home page" of the service, click on hypertext links, retrieve some information, fill in and post HTML forms, obtain another piece of information, and so on. The various pieces of information located on one page can be: in a pre-structured format, in a semi-structured format, or in unstructured plain text.

By pre-structured format, we mean a format which is known in advance by the user. This is typically the case of pages using a data representation compliant with a standard, such as the Open Financial Connectivity standard. Where information producers are able to agree on such standard representations COIN can take advantage of the format guarantees.

Semi-structured format includes data presented in a table, a list, a tree, or other structuring organization, but for which the structure is not fully know in advance and must be parsed and analyzed on the fly to locate the data. There are a large number of information sources on the World Wide Web (e.g., CIA fact book, stock exchange quote services, weather reports and weather forecasts, etc.) offering semi-structured format data.

The COIN Web-wrapping technology is based on a high level declarative language for the specification of wrapper interface and actions. This language specifies what information can be extracted from a source. The generic wrapper engine transforms user requests into a plan for extracting the relevant data according to the specification, executes the plan by accessing the source, and organizes and presents the extracted data. The specification language for the generic Web-wrappers allows the definition of a state transition network. The transitions in the network correspond to the hyperlinks in the hypertext, additionally, the information initially inputted or collected in the preceding stages is carried and is used to define the transitions, fill the parameters of a form, or choose a link among several on a page. On each page (or state of the transition network), the Web-wrapper specification uses patterns (e.g., regular expressions) to identify the location of data to be extracted, input fields for a form, and links to other locations. More recently we have moved beyond regular expression patterns so that we can take advantage of the structure of information on a page as provided by XML tags.

Furthermore, web sites have differing capabilities. Some sites are collections of static pages, others are dynamically created pages based upon specific interactions. It is necessary to take into consideration the specific capabilities and limitations on data retrieval from sites.

3. Mediation.

In a heterogeneous and distributed environment, the mediator transforms a query written in the terms known to the user or application program (i.e., according to the user's or programmer's assumptions and knowledge) into one or more queries in the terms of the component sources. The individual subqueries may still involve several sources. Subsequent planning, optimization and execution phases are needed. Typically, the planning and execution phases will consider the limitations of the sources and the topology and costs of the network. The execution phase is in charge of the scheduling of the query execution plan and the realization of the complementary operations that could not be handled by the sources individually (e.g. a join across sources).

The first mediation phase can be naively described as the rewriting of the query against a "view definition", the view of the disparate information sources that the mediation service provides to the user or application program. The main quality of the mediation approach will depend on its properties with respect to the strategy for the assimilation and definition of the knowledge needed for the construction of this "view definition." Where a large number of independent information sources are accessed (as is now possible with the global information infrastructure), flexibility, scaleability, and non-intrusiveness will be of primary importance.

Traditional tight-coupling approaches to semantic interoperability rely on the *a priori* creation of federated views on the heterogeneous information sources. Although they provide good support for data access, they do not scale-up efficiently given the complexity involved in constructing and maintaining a shared schema for a large number of, possibly independently managed and evolving, sources. Loose-coupling approaches rely on the user's intimate knowledge of the semantic conflicts between the sources and the conflict resolution procedures.

This flexibility becomes a drawback for scaleability when this knowledge grows and changes as more sources are join the system and when sources are changing.

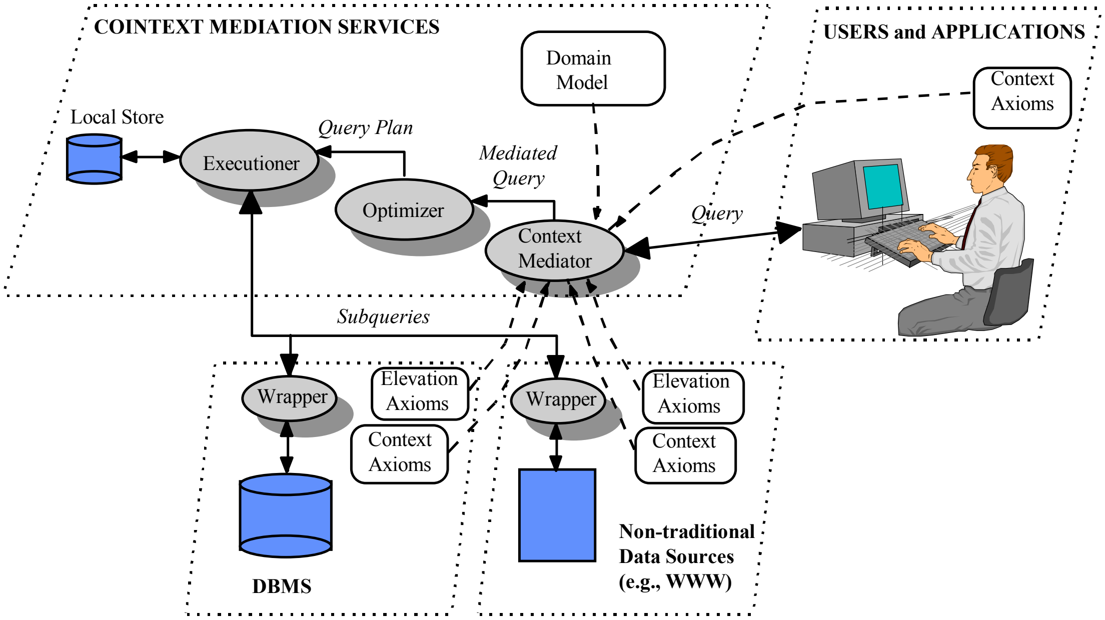

Figure 2. The Architecture of the Context Interchange System

The Context Interchange (COIN) approach is a middle ground between these two approaches. It allows queries to the sources to be mediated, i.e. semantic conflicts to be identified and solved by a context mediator through comparison of contexts associated with the sources and receivers concerned by the queries. It only requires the minimum adoption of a common Domain Model which defines the domain of discourse of the application.

The knowledge needed for integration is formally modeled in a COIN framework [Goh96], The COIN framework is a mathematical structure offering a sound foundation for the realization of the Context Interchange strategy. The COIN framework comprises a data model and a language, called COINL, of the Frame-Logic (F-Logic) family [KLW95, DoT95]. The framework is used to define the different elements needed to implement the strategy in a given application:

- The Domain Model is a collection of rich types (semantic types) defining the domain of discourse for the integration strategy;
- Elevation Axioms for each source identify the semantic objects (instances of semantic types) corresponding to source data elements and define integrity constraints specifying general properties of the sources;
- Context Definitions define the different interpretations of the semantic objects in the different sources or from a receiver's point of view.

The Domain Model, the different sets of Elevation Axioms, the Context Definitions, together with additional generic axioms defining the mediation, constitute a COINL program. This program controls the query mediation engine.

Let us consider a simple example where a user issues the query Q1 to a source called "security" providing historical financial data about a stock exchange. The user and the source have different assumptions regarding the interpretation of the data. These assumptions are captured in their respective contexts C1 and C2. The Domain Model defines semantic types such as money amounts, dates, and company identifications. Query Q1 requests the price of the IBM security on March 12, 1995:

```
(Q1)    select security.Price
            from security
            where security.Ticker = "IBM"
           and security.Date = "12/03/95";
```

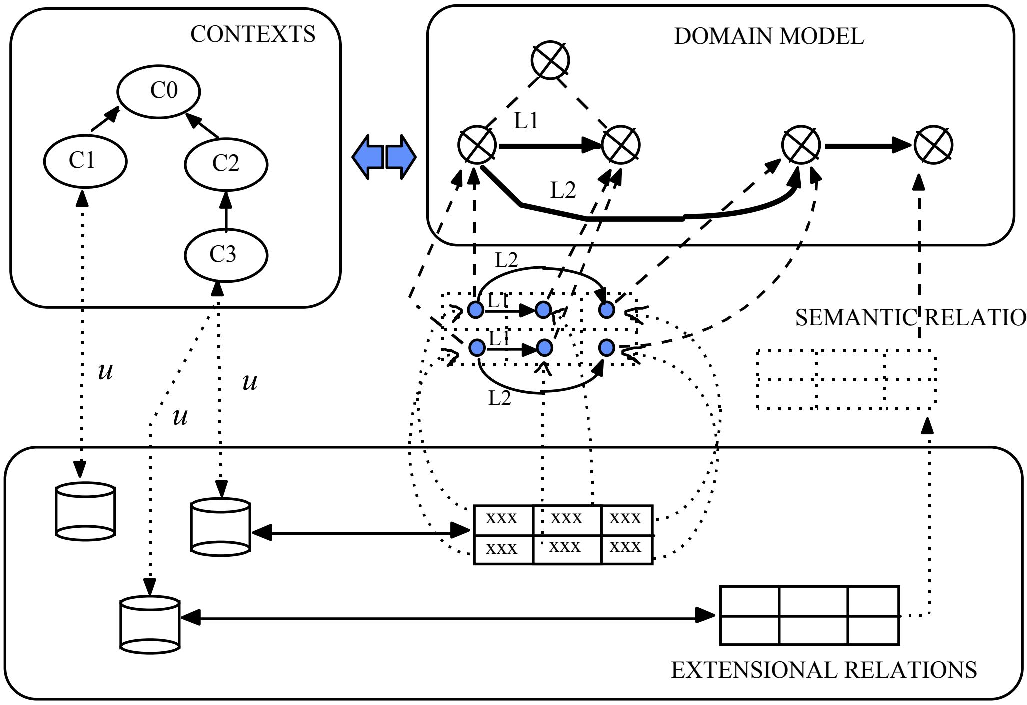

Figure 3. The Context Interchange Formal Framework

The receiver's context C1 assumes money amounts are in French Francs, dates in the European format, and that currency conversions should use the date of the money amount validity. We see immediately that context information is needed to avoid the confusion between March 12 and December 3, 1995. On the other hand, the source context C2 expresses its money amounts in the local currencies of the company, and dates are in American format. The mediation rewrites the query, incorporating the proper currency conversion (as of March 12, 1995) making use of an ancillary source called "cc" for exchange rates, and the proper date format conversion. The resulting mediated query MQ1 is:

```
(MQ1) select security.Price * cc.Rate
            from security, cc
            where security.Ticker = "IBM"
              and security.Date = "03/12/95"
              and cc.Expressed = "USD"
              and cc.Exchanged = "FRF"
              and cc.AsOfDay = security.Date;
```

In this example, the domain model will define the various semantic types corresponding to the concepts associated to the data elements manipulated in the application domain. For instance, semantic types capturing notions like money amounts, company financials or exchange rates need to be defined. If some relationships exist among these semantic types and are relevant from an ontological point of view (as opposed to the peculiarities of the structures hosting the data in the sources), they can be represented in the domain model by means of attributes. The following is an excerpt of the domain model of our example in COINL[1] :

```
moneyAmount: number;
companyFinancials: moneyAmount;
exchangeRate: number [   to  ⇒ currency;
                         from ⇒ currency;
                         asof ⇒ date].
```

The elevation axioms define the semantic image of the relations and the data exported by the sources. Below is an excerpt of the elevation axioms for a source exporting a relation **Olsen** reporting historical data for currency exchange rates (Olsen is an actual Web site, which can be utilized as if it were a relational database through use of COIN's Web Wrapping technology). The first rule defines the semantic relation **Olsen_semantic.** The second rule, defines an exchangeRate semantic object. The third rule is an integrity constraint expressing the reversibility of the rate.

```
Olsen_semantic(     f_to(To, From, Date),
                    f_from(To, From, Date),
                    f_date(To, From, Date),
                    f_rate(To,From, Date)) ←
                              olsen(To, From, Date, Rate).
f_rate(To, From, Date): exchangeRate
               [to ⇒ f_to(To, From, Date),
                from ⇒ f_from(To, From, Date),
                date ⇒ f_date(To, From, Date)].
Olsen(To, From, Date, Rate1), Olsen(From, To, Date, Rate2)→
                         Rate1 = 1/Rate2.
```

The context associated with the sources and the receivers define the *modifiers* of the semantic objects. The modifiers are special attributes dependent on the context and determine the interpretation of the data. They are used for the identification of conflicts during the query mediation. They can be defined by extension (given a value) or by intention (by means of a rule). Several modifiers corresponding to different notions determining the interpretation of a semantic object are associated to it (e.g., the currency and the as-of date of a money amount). Modifiers are declared for all objects of a given semantic type.

```
X:moneyAmount
     [[currency.value ⇒  "FRF"];
      [asofdate ⇒ V] →  X[report.date ⇒ V] ].
```

Finally, the conversion functions for each modifier locally defines the resolution of potential conflicts. The conversion functions can be defined in COINL but are likely, in practical cases, to rely on external services or external procedures. The relevant conversion functions are

[1] In this document we are using the abstract syntax of COINL in order to give the reader an intuition of the logical constructs in the language. End-users and programmers are offered visual or graphical interfaces and a concise concrete syntax (of the family of OQL).

gathered and composed during mediation to resolve the conflicts. No global or exhaustive pairwise definition of the conflict resolution procedures is needed.

Both the query to be mediated and the COINL program are combined into a definite logic program (a set of Horn clauses) where the translation of the query is a goal. The mediation is performed by an abductive procedure which infers from the query and the COINL programs a reformulation of the initial query in the terms of the component sources. The abductive procedure makes use of the integrity constraints in a constraint propagation phase which has the effect of a semantic query optimization. For instance, logically inconsistent rewritten queries are rejected, rewritten queries containing redundant information are simplified, rewritten queries are augmented with auxiliary information.

Although the procedure itself is inspired by the Abductive Logic Programming framework [KKT93] and can be qualified as an abduction procedure, we do not argue that abduction by itself is a suitable philosophical concept for mediation, but rather take advantage of formal logical framework for the study and implementation of an appropriate procedure. One of the main advantages of the abductive logic programming framework is the simplicity in which it can be used to formally combine and to implement features of query processing, semantic query optimization and constraint programming.

The COIN abductive framework can also be extrapolated to problem areas such as integrity management, view updates and intensional updates for databases. Because of the clear separation between the declarative definition of the logic of mediation into the COINL program from the generic abductive procedure for query mediation, we are able to adapt our mediation procedure to new situations such as mediated consistency management across disparate sources, mediated update management of one or more database using heterogeneous external auxiliary information or mediated monitoring of changes. Although there are fundamental theoretical limits in many areas, such as view update, we can extend the range of mediation services to handle a broader range of client needs.

The mediated update problem illustrates the potential advantage of the formal logical approach in COIN over traditional view mechanisms for mediation. For a retrieval, either approach can be made to deliver correct results (with more or less effort). The COIN approach, however, holds the knowledge of the semantics of data in each context and across contexts in declarative logical statements separate from the mediation procedure. An update asserts that certain data objects must be made to have certain values in the updater's context. An update mediation algorithm by combining the update assertions with the COIN logical formulation of context semantics, can determine whether is unambiguous and feasible, and if so, what source data updates must be made to achieve the intended results. If ambiguous or otherwise infeasible, the logical representation may be able to indicate what additional constraints would clarify the updater's intention sufficiently to the update to proceed.

We are also applying the COIN framework to important aspects of the source selection problem. Integrity constraints in COINL and the consistency checking component of the abductive procedure provide the basic ingredients to characterize the scope of information available from each source, to efficiently rule out irrelevant data sources and thereby speed up the selection process. For example, a query requesting information about *companies with assets lower than $2 million* can avoid accessing a particular source based on knowledge of integrity constraints stating that *the source only reports information about companies listed in the New York Stock Exchange* (NYSE), and that *companies must have assets larger than $10 million to be listed in the NYSE*. In general, integrity constraints express necessary conditions imposed on data. However, more generally, a notion of completeness degree of the domain of the source with respect to the constraint captures a richer semantic information and allows more powerful source selection. For instance, a source could contain exactly or at least all the data verifying the constraint (information about all the companies listed in the NYSE are exhaustively reported in the source).

Conclusion

We are in the midst of exciting times – the opportunities to make use of diverse information sources are incredible but the challenges are considerable. The effective use of metadata can enable us to overcome the challenges and more fully realize the opportunities. A particularly interesting aspect of the context mediation approach described is the use of metadata to describe the expectations of the receiver as well as the semantics assumed by the sources. If we do not address these challenges directly and effectively, we might endure serious consequences, as illustrated by the historical example displayed in the box below.

The 1805 Overture

In 1805, the Austrian and Russian Emperors agreed to join forces against Napoleon. The Russians promised that their forces would be in the field in Bavaria by **Oct. 20**.

The Austrian staff planned its campaign based on that date in the **Gregorian calendar**. Russia, however, still used the ancient **Julian calendar**, which lagged 10 days behind.

The calendar difference allowed Napoleon to surround Austrian General Mack's army at Ulm and force its surrender on Oct. 21, well before the Russian forces could reach him, ultimately setting the stage for Austerlitz.

Source: David Chandler, *The Campaigns of Napoleon*, New York: MacMillan 1966, pg. 390.

ACKNOWLEDGEMENTS

Work reported herein has been supported, in part, by the Advanced Research Projects Agency (ARPA) and the USAF/Rome Laboratory under contract F30602-93-C-0160, the MIT International Financial Services Research Center (IFSRC), the MIT Leaders For Manufacturing (LFM) Program, the MIT PROductivity From Information Technology (PROFIT) Program, and the MIT Total Data Quality Management (TDQM) Program. Information about the Context Interchange project can be obtained at http://context.mit.edu.

REFERENCES

[BFG96a] Bressan, S., Fynn, K., Goh, C.H., Jakobisiak, M., Hussein, K., Kon, H., Lee, T., Madnick, S., Pena, T., Qu, J., Shum, A., and Siegel, M. (1996). "The COntext INterchange mediator prototype," *SIGMOD97*.

[BFG96b] Bressan, S., Fynn, K., Goh, C.H., Madnick, S., Pena, T., and Siegel, M. (1996). "Overview of a prolog implementation of the COntext INterchange mediator," *Practical Applications of Prolog 97*.

[DaGH95] Daruwala, A., Goh, C.H., Hofmeister, S., Hussein, K., Madnick, S., and Siegel, M. (1995). "The context interchange network prototype," In *Proc of the IFIP WG2.6 Sixth Working Conference on Database Semantics (DS-6)*, Atlanta, GA. To appear in LNCS (Springer-Verlag).

[DoT95] Dobbie, G. and Topor, R. (1995). "On the declarative and procedural semantics of deductive object-oriented systems," *Journal of Intelligent Information Systems*, 4:193--219.

[Goh96] Goh, C. (1996). *Representing and Reasoning about Semantic Conflicts* In *Heterogeneous Information System*, PhD Thesis, MIT.

[GBMS96a] Goh, C.H., Bressan, S., Madnick, S.E., and Siegel, M.D. (1996). "Context Interchange: Representing and reasoning about data semantics in heterogeneous systems," *Sloan School Working Paper #3928*, Sloan School of Management, MIT, 50 Memorial Drive, Cambridge MA 02139.

[GBMS96b] Goh, C.H., Bressan, S., Madnick, S.E., and Siegel, M.D. (1996). "Context Interchange: New Features and Formalisms for the Intelligent Integration of Information," *Sloan School Working Paper*, Sloan School of Management, MIT, 50 Memorial Drive, Cambridge MA 02139.

[GMS94] Goh, C.H., Madnick, S.E., and Siegel, M.D. (1994). "Context interchange: overcoming the challenges of large-scale interoperable database systems in a dynamic environment," In *Proc of the Third International Conference on Information and Knowledge Management*, pages 337--346, Gaithersburg, MD.

[Jako96] Jakobisiak, M. (1996). "Programming the web -- design and implementation of a multidatabase browser," *Technical Report CISL WP #96-04*, Sloan School of Management, Massachusetts Institute of Technology.

[KKT93] Kakas, A.C., Kowalski, R.A., and Toni, F. (1993). "Abductive logic programming," *Journal of Logic and Computation*, 2(6):719--770.

[KLW95] Kifer, M., Lausen, G., and Wu, J. (1995). "Logical foundations of object-oriented and frame-based languages," *JACM*, 4:741—843.

[Qu96] Qu, J.F. (1996). "Data wrapping on the world wide web," *Technical Report CISL WP #96-05*, Sloan School of Management, Massachusetts Institute of Technology.

[ScSR94] Sciore, E., Siegel, M., and Rosenthal, A. (1994). "Using semantic values to facilitate interoperability among heterogeneous information systems," *ACM Transactions on Database Systems*, 19(2):254--290.

[Shum96] Shum, A. (1996). *Open Database Connectivity of the Context Interchange System*, Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science.

[SM91a] Siegel, M. and Madnick, S. (1991). "Context Interchange: Sharing the Meaning of Data," *SIGMOD RECORD*, Vol. 20, No. 4, December p. 77-8.

[SM91b] Siegel, M. and Madnick, S. (1991). "A metadata approach to solving semantic conflicts," In *Proc of the 17th International Conference on Very Large Data Bases*, pages 133--145.