Context Interchange Mediation for Semantic Interoperability and Dynamic Integration of Autonomous Information Sources in the Fixed Income Securities Industry

Allen Moulton Stuart E. Madnick Michael D. Siegel
MIT Sloan School of Management
amoulton@mit.edu smadnick@mit.edu msiegel@mit.edu

Abstract

We examine semantic interoperability problems in the fixed income securities industry and propose a knowledge representation architecture for context interchange mediation to support dynamic integration of autonomous database, web, and procedural sources of information. For sources and receivers sharing a common subject domain, the mediator's reasoning engine can devise query plans integrating multiple sources and resolving semantic heterogeneity. Receiver applications can obtain the data they need in the form they need it without imposing changes on sources. The architecture includes: 1) data models for each source and receiver, 2) subject ontologies, containing abstract subject matter conceptualizations that would be known to experienced practitioners in the industry, and 3) context models for each source and receiver that explain how each data model implements the abstract concepts from a subject ontology.

1. Introduction

Efficiently integrating new sources of information from outside the enterprise is often critical to success in a world of global competition, interdependency, and rapid market change. Within an organization, data is created, stored, and used by people and computers sharing a common implicit understanding of data semantics. We use the term *context* to refer to this implicit understanding of the relationship between data elements and structures and the real world that the data represents. The context interchange problem arises when organizations with different contexts must exchange information[8].

A context interchange (COIN) mediator is an automated reasoning engine to assist an organization in resolving semantic conflicts between its own receiver context and the contexts of data sources[6]. Because context definitions are declarative, they need only be prepared once for each source and receiver context[1]. Data sources may be relational databases, XML documents, HTML webs wrapped to appear as relations with limited query capability[5], and stateless computational procedures. Using declarative context knowledge, a COIN mediator identifies semantic conflicts and designs plans for combining sources with data conversions to meet receiver semantic requirements.

Sources and receivers are seen as autonomous implementations of common subject domain abstractions. Source and receiver system designers make decisions about how to conceptualize abstract constructs and about how to represent that conceptualization in data and programs. The COIN mediator has the task of applying declarative information about the context of each source and receiver to devise plans for integrating sources to meet receiver requirements.

Given a large number of component systems operating in a diversified and dynamic environment, COIN mediation facilitates: rapid incorporation of new information sources, dynamic substitution of information sources, extension and evolution of semantics, data representation in the user's context, access to the meaning of data represented, identification and selection of information source alternatives, and adaptation to changes in user and business operations.

Building on earlier work by Goh[6], new knowledge representation and reasoning methods expand the functionality of COIN mediation to include: 1) identifying data representation conflicts and introducing conversions to transform data from source to receiver form, 2) applying subject domain and context knowledge to map between receiver schema and source schemata, 3) determining when and how to combine sources, feeding data from one source to another with appropriate data conversions, 4) deriving missing data by applying domain ontology, context knowledge, or by combining sources.

2. Fixed income securities industry

Information is the critical resource in the fixed income securities industry – information about securities and their issuers, information about markets, information about economic conditions and events, and information about methodologies and models. Billions of dollars of debt instruments trade every week. Firms on the "buy side" (institutional investors and investment managers) manage capital on behalf of investors. Firms on the "sell side" (investment banks, brokers, and dealers, often known as "Wall Street") brings new securities to market and interacts to create capital markets. With relatively low capitalization compared to the capital invested by buy-side firms, sell-side make money from small commissions or price spreads on large volumes of transactions. The sell-side offers buy-side managers access to its skill, expertise, and knowledge in anticipation of purchase and sale orders.

Bonds and other fixed income securities are obligations to pay sums of money at points in time over the life of the security. To an investor, a fixed income security represents a stream of future cash flows. There may be optional events that change the cash flow stream and there may be risk of default. In essence, however, all fixed income securities are interchangeable commodities from the point of view of an investor. The cash flows from one or more obligations may even be repackaged by selling off rights to payments or by combining rights to payments into new composite securities. This repackaging, or "financial engineering" can produce securities known as "derivatives." Faced with a vast array of combinations of cash flows, risks, and optional events, every industry participant needs timely information and effective methods for determining investment value from raw data[2].

2.1 Portfolio manager requirements

Fixed income portfolio managers may need to draw upon external sources for data about security characteristics, for market valuation information, and for models and calculations[10]. All these sources may need to be combined with internal portfolio holdings data and accessed through a decision support application system. Table 1 shows a partial schema and semantics requirements for an offerings analysis application that obtains current dealer offerings and presents the portfolio manager information about securities offered and dealer prices in a consistent form. The first three attributes provide information about the security offered (standard CUSIP identifier, maturity date, and coupon interest rate). The last attribute is the price asked by the dealer, expressed as a percentage of face value with fractions in 32nds (e.g., 116 8/32 in Table 1). A fragment of a query by the offerings application might be:

Receiver relation R (application requirements)		
attribute	**sample data**	**semantic notes**
secidn	191219AN4	CUSIP security identifying number
matdat	02/01/2012	maturity date, mm/dd/yyyy
cprate	8.500	interest rate, percent, decimal fraction
price	116.08	dollar price, percent, fraction in 32nds

Table 1. Receiver data semantics

```
SELECT secidn, matdat, cprate, price   FROM R  WHERE <criteria>
```

To explore context interchange problems, we consider two alternative sources for offerings. Dealer A provides a web page that has been wrapped for a relational query interface[5]. Dealer B provides an XML document. Each dealer organization makes its own decisions about what information to present and the semantics of that information.

2.2 Data representation semantics

Table 2 shows a section of the schema for Dealer A offerings. The first step in developing a mediation plan is to note that each row in source A matches the requirement for a row in R – each represents a dealer offering for a security. From the semantic notes, it is clear that, although three

Source relation A (dealer A offerings web page)		
attribute	**sample data**	**semantic notes**
cusip	191219AN4	CUSIP security identifying number
maturity	40940	maturity date, Lotus/Excel 1900 date
coupon	0.08500	interest rate, factor, decimal fraction
price	116.25	dollar price, percent, decimal fraction

Table 2. Dealer A data semantics

attribute names are different, each attribute in R can be obtained from one attribute in A.

The final step is to resolve data representation differences. R.secidn and A.cusip are the same. R.matdat requires a date in "mm/dd/yyyy" format; A.maturity is provided as a Lotus date sequence number. R.cprate is in percent; A.coupon is in factor form commonly used in spreadsheets. The price attributes, though named the same, are subtly different in semantics. R.price requires a percent with fraction in 32nds; A.price is expressed as a percent with decimal fraction. Failure to convert from decimal to 32nds could result in a substantial error in the price. Having identified the semantic conflicts, the mediator inserts appropriate data conversions: multiplying by 100 to convert factor to percent, the Excel "dollarfr" function to convert a decimal fraction into 32nds, and a wrapped date conversion function, source F (see Table 3).The query rewritten in terms of the source schema with necessary data representation conversions would be:

Source F (wrapped date conversion function)		
attribute	**sample data**	**semantic notes**
out	02/01/2012	reformatted date output
outformat	"mm/dd/yyyy"	format for output date
in	40940	date input
informat	"Lotus"	format for input date

Table 3. Date conversion data semantics

```
SELECT cusip as secidn, F.out as matdat, coupon*100 as cprate, dollarfr(price,32) as price  FROM A, F
 WHERE <criteria>    AND F.outformat = "mm/dd/yyyy"    AND F.in = maturity AND F.informat = "Lotus"
```

2.2 Derived data and multiple source integration

The use of XML simplifies the access to data in many respects, but still leaves a wide range of semantic issues to be resolved. Adoption of standards can reduce the degree of semantic heterogeneity. Nevertheless, in the securities industry and many others, innovation will proceed faster than standards. Consider Dealer B offerings provided as an XML document (*viz*. Fig. 1).

```
<OFFERSHEET>    <OFFER>
    <BOND> 191219AN4 </BOND>
    <PRICE> 103.28 </PRICE>
  </OFFER>  ...  </OFFERSHEET >
```

Figure 1. Dealer B Offerings

Dealer B offerings have a tabular structure that can be viewed as a relation as shown in Table 4. Comparing the semantics of B to requirements R, we note that two of the attributes of the security are missing. Furthermore, the price is expressed as a "nominal spread" in "basis points" instead of a "dollar price" in percent. To meet the receiver's requirements, general industry knowledge and additional data sources must be brought to bear, along with conversion of units and scaling.

Source relation B (dealer B XML document)		
attribute	**sample data**	**semantic notes**
BOND	191219AN4	CUSIP security identifying number
PRICE	103.28	nominal spread, basis points

Table 4. Dealer B XML sample and semantics

To resolve the semantic conflict, the mediator must know 1) source C can provide security details, 2) nominal spread means the difference between yield on a security and a benchmark yield, 3) the on-the-run 10-year T-note yield is an appropriate benchmark, 4) which can be obtained from source D, 5) bond calculation source object E can convert yield to price given the security's interest rate and other details, 6) rules for converting data codes, 7) basis points are 1/100th of a percent, and 8) methods for converting data representations as discussed above.

Source relation C (security characteristics web site)		
attribute	**sample data**	**semantic notes**
coupon	8.500	interest rate, percent, decimal fraction
maturity	02-01-2012	maturity date, MM-DD-YYYY
cusip	191219AN4	CUSIP security identifying number
datedDate	02-11-1992	issue date, MM-DD-YYYY
firstCoup	08-01-1992	first payment date, MM-DD-YYYY
market	US Corporate	market/type of security, text
payFreq	Semi-Annual	interest payment interval

Table 5. Source C data semantics

After analyzing the semantic differences between the receiver R and source B, the mediator identifies additional data sources C (Table 5) and

Source relation D (Treasury yield curve web site)		
attribute	**sample data**	**semantic notes**
10yr	5.091	yield on current 10 year T-note

Table 6. Source D data semantics

D (Table 6). The mediator also inserts a necessary yield-price calculation function, specified as if it were a data source E (Table 7).

The mediator must also insert data conversions for dates and percentages as described above (section 2.2). Data codes for payment frequency from source C must be mapped to E's context and the day count basis inferred from market conventions. Combining these sources, data conversions, mappings, and inferences, the resultant mediated query would be:

Source E (Excel analytic toolkit function PRICE)		
attribute	**sample data**	**semantic notes**
price	117.875	flat price, percent, decimal fraction
settlement	37196	settlement date, Excel 1900 date
maturity	40940	maturity date, Excel 1900 date
rate	0.0850	interest rate, factor, decimal fraction
yld	0.061238	yield, factor, decimal fraction
redemption	100	redemption value, percent
frequency	2	coupon frequency per year (1,2,4)
basis	0	day count basis, code (0,1,2,3,4)

Table 7. Source E data semantics

```
SELECT B.cusip as secidn, v.out as matdat,
    C.coupon as cprate, dollarfr(E.price,32) as price
  FROM B, C, D, E, F v, F w, F x, Cfreq, Cmarket, Efreq, Ebasis, Mmarket
  WHERE <criteria> AND C.cusip = B.BOND AND v.in = C.maturity AND v.infmt = "mm-dd-yyyy" AND v.outfmt = "mm/dd/yyyy"
    AND E.settlement = x.out AND x.in = "11/01/2001" AND x.outfmt = "Lotus" AND x.infmt = "mm/dd/yyyy"
    AND E.maturity = w.out AND w.in = C.maturity AND w.outfmt = "Lotus" AND w.infmt = "mm-dd-yyyy"
    AND E.rate = C.coupon/100  AND E.yld = ( B.PRICE/100 + D.10yr ) / 100 AND E.redemption = 100
    AND E.frequency = Efreq.xcode  AND E.basis = Ebasis.xcode  AND Efreq.freq = Cfreq.freq AND C.payFreq = Cfreq.xcode
    AND C.market = Cmarket.xcode AND Mmarket.daycount = Ebasis.daycount
```

Without mediation, the portfolio manager would see a price of 116.25 from Dealer A and 103.28 from Dealer B. With mediation, Dealer B's quote is converted to a dollar price of 117 28/32 and the choice is reversed.

3. Knowledge representation architecture

Our knowledge representation architecture divides the knowledge used for mediation into three layers: 1) a domain ontology containing abstract subject domain concepts used by experienced practitioners and system designers in the industry, 2) data models for each source and receiver with the kind of information programmers would use to access data, and 3) context models for each source and receiver that explain how each source or receiver data model implements the abstract concepts from a subject domain ontology.

The framework of a subject domain ontology (*viz*. Figure 2) is a structural conceptual model with classes of abstract objects, attributes of objects, and relationships. Semantic types capture alternative data representations [6]. Enumerated conceptual categories model object property distinctions which may be implemented with different symbolic codes by each source and receiver. Rules capture functional relationships among conceptual model attributes that would be known from general domain knowledge. Default and contingent rules allow for deriving attributes based on partial information following the reasoning that industry participants would use.

Data models for the receiver and for relational sources use schema and catalog information. For XML, an XML schema or DTD can be used or the schema inferred from documents themselves. For HTML sources, the data model is provided by the web wrapper. For computational procedural sources, arguments and return values are treated as relational attributes in a data model that is augmented with functional dependency and input-output combination constraints.

Context models for each source and receiver explain how each data model implements the general concepts in the domain ontology. Classes from the domain ontology conceptual model can be used directly or augmented with context-specific extensions. Context-specific functional or equivalence relationships tie elements of the conceptual model to elements of the data model. For coding schemes, enumerated attribute domains are mapped to conceptual categories from the domain ontology [11]. Semantic types are used to logically encapsulate data attributes and associate context-specific modifier values to identify the data representation used [6].

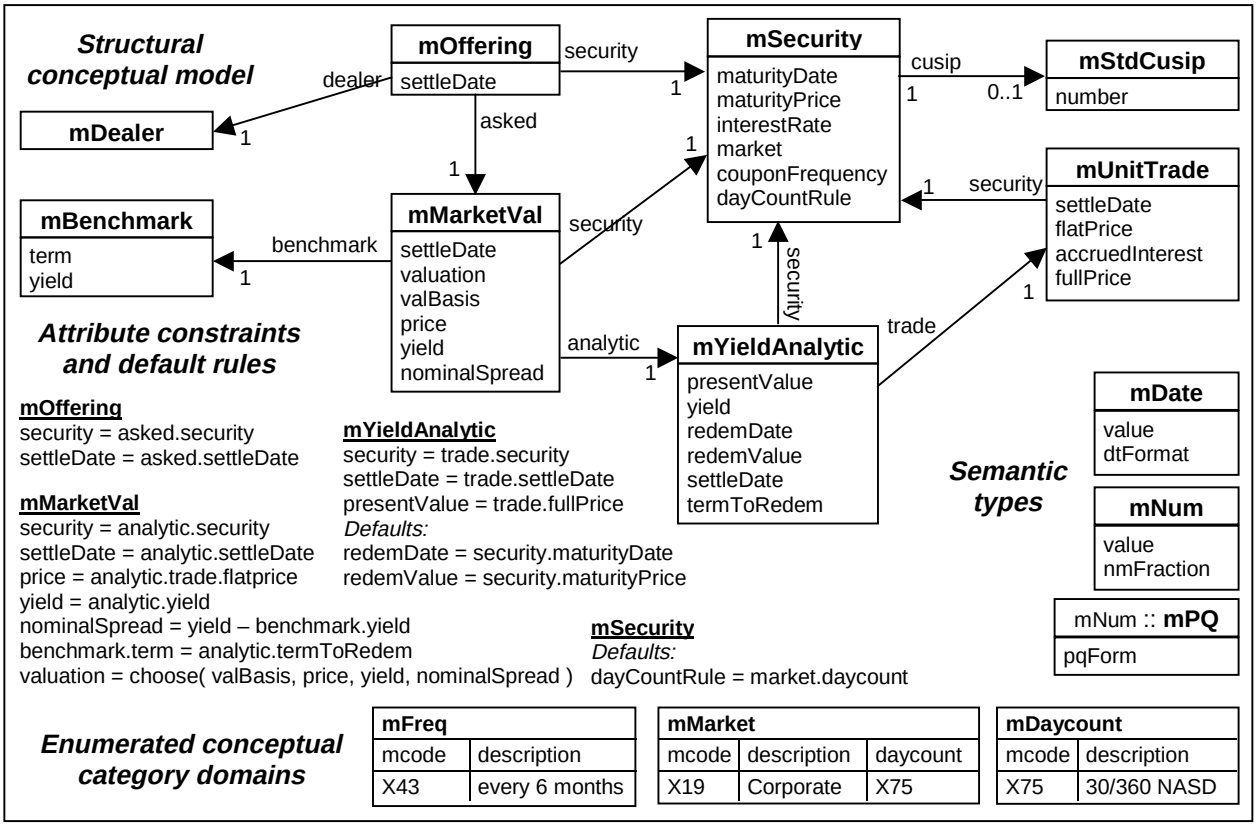

Figure 2. Sample fragment of a subject domain ontology for fixed income securities

Although similar in appearance to a a global schema, a subject domain ontology serves as an abstract representation of the subject matter that each source and receiver data model implements in its own way. Neither sources nor receivers need to accept the domain ontology as the "right way" of representing information about the subject matter at hand, avoiding some of the practical user acceptance problems noted in [9]. By allowing each context model to extend the domain ontology and to explain how context-specific concepts map to general domain ontology concepts, semantic interoperability is facilitated without imposing the rigidity seen in view-based systems.

Defining context using our architecture is analogous to the process that a programmer would follow to design a program to extract data from sources. The first step is to model each source or receiver relation using conceptual objects from the domain ontology. In the example above, rows of R are modeled by the ontology *mOffering* concept. Next, each data attribute in R is associated with a conceptual attribute from the ontology (Table 8), modeled with a semantic type with modifiers to specify the data representation used in the context.

attribute in R	path from mOffering concept
secidn	security.cusip.number
matdat	security.maturityDate
cprate	security.interestRate
price	askVal.valuation
	askVal.valBasis=price

Table 8. Context model fragment for R

Sources A and B would be modeled similarly. Source C would be modeled with a *mSecurity* object. The one-to-one relationship between a *mStdCusip* and a *mSecurity* allows the mediator to associate the *mSecurity* implicitly referred to in B with the data available from source C. Attributes *C.payFreq* and *E.frequency* represent the same concept using context-specific symbols. In defining a context, tables of symbols or codes are gathered from documentation or usage and then mapped to conceptual categories from the ontology (Table 9), by which the mediator can convert a C code to an E code when the sources need to be joined [see 11 for details].

Cfreq		Efreq	
xcode	freq	xcode	freq
Semi-Annual	X43	2	X43

Table 9. Context code mappings to conceptual categories

Source D is modeled with an *mBenchmark* object, while source E implements the *mYieldAnalytic* concept. By using the structural conceptual model and attribute constraint rules, the mediator can reason about how the sources can be fit together with each other to derive the values required by the receiver. As a final step the mediator identifies data representation conflicts by matching modifiers on semantic types and inserting conversion functions associated with the semantic types to produce a plan such as the SQL at the end of section 2.3.

4. Conclusion

Context interchange mediation brings automated methods to the important task of assuring that data exchanged across organizations can meet the semantic requirements of the receiver – and do so without obligating source organizations to change their way of doing business or to know about or accommodate the needs of the receiver. We are exploring techniques for specifying contexts and ontologies using well established methodologies of business systems analysis and database design [*viz*. 3, 4].

The evolving ISO TC68/SC4/WG10 [12], securities industry standards may partially solve the interoperability problem, and should additionally provide detailed substantive information models for building ontologies and context models for a wide range of securities information interchange requirements [10,11]. Context interchange mediation can play an important role in resolving semantic interoperability problems arising from aggregation and web services[7].

Ongoing research includes the specification of logic rules and reasoning algorithm to traverse the analytic process from receiver data model requirements, through receiver context models and subject domain ontologies, thence to potential source context models and data models, devising plans for meeting the receiver's needs from available source data combined with generated conversion relations. Future work includes: contextual conflicts among interrelated data within a source and across multiple sources, extended integration of semi-structured, unstructured and image data sources, domain ontology and context model development and evolution methodologies and tools, interlocking subject domain integration, automatic source selection, and extensions to the reasoning paradigm to incorporation of complex object components, such as those described in [2].

References

[1] S. Bressan, C. H. Goh, N. Levina, S. E. Madnick, A. Shah and M. D. Siegel. "Context Knowledge Representation and Reasoning in the Context Interchange System," Applied Intelligence (13:2), Sept. 2000, pp. 165-179.

[2] Rakesh Chandra and Arie Segev: "Managing Temporal Financial Data in an Extensible Database." VLDB 1993: pp. 302-313.

[3] Stephen Cranefield and Martin K. Purvis. "UML as an Ontology Modelling Language," Proc. Workshop on Intelligent Information Integration, IJCAI 1999.

[4] Ramez Elmasri, Shamkant B. Navathe. Fundamentals of Database Systems, 3rd Edition. Addison-Wesley, 2000.

[5] A. Firat, S. E. Madnick, and M. D. Siegel. "The Caméléon Web Wrapper Engine," Proc. VLDB 2000 Workshop on Technologies for E-Services, Sept., 2000.

[6] C. H. Goh S. Bressan., S. E. Madnick and M. D. Siegel "Context Interchange: New Features and Formalisms for the Intelligent Integration of Information," ACM Trans. on Office Information Systems, July 1999, pp 270-293.

[7] Mark Hansen, Stuart E. Madnick, Michael Siegel: Data Integration using Web Services. DIWeb 2002: 3-16.

[8] S. E. Madnick. "Metadata Jones and the Tower of Babel: The Challenge of Large-Scale Heterogeneity," Proc. IEEE Meta-Data Conf., April 1999.

[9] A. Moulton, S. Bressan, S. E. Madnick and M. D. Siegel. "An Active Conceptual Model for Fixed Income Securities Analysis for Multiple Financial Institutions," Proc. ER 1998.

[10] A. Moulton, S. E. Madnick and M. D. Siegel. "Context Mediation on Wall Street," Proc. CoopIS 1998, pp. 271-279.

[11] A. Moulton, S. E. Madnick and M. D. Siegel. "Semantic Interoperability in the Securities Industry: Context Interchange Mediation of Semantic Differences in Enumerated Data Types," Proc. DEXA WEBH 2002.

[12] SWIFT. "ISO Working Group 10," http://www.swift.com/index.cfm?item_id=6610