COntext INterchange (COIN) System Demonstration

Aykut Firat (aykut@mit.edu)
M. Bilal Kaleem (mbilal@mit.edu)
Philip Lee (philee@mit.edu)
Stuart Madnick (smadnick@mit.edu)
Allen Moulton (amoulton@mit.edu)
Michael Siegel (msiegel@mit.edu)
Hongwei Zhu (mrzhu@mit.edu)

MIT Sloan School of Management
30 Wadsworth Street, Cambridge, MA
02142, USA
http://context2.mit.edu/coin

Abstract. The Context Interchange (COIN) System provides tools for representing, processing, and reconciling heterogeneous data semantics. In this demonstration we show how COIN can be used to automatically resolve semantic conflicts. We demonstrate support tools for developing COIN-compatible applications and show the representation and resolution capabilities in COIN. We then show how the domain application merging capabilities in COIN allow us to rapidly develop new applications which combine the domain models, context, and elevation axioms of existing applications. This is done without rewriting existing domain knowledge. Instead a tool is used to create linking axioms. We demonstrate the construction of a Travel Agent application by merging existing airfare and car rental applications. The new application combines the strength of both application domains and resolves their semantic differences.

1 Introduction

Context Mediation technology addresses the important problem of data interpretation and deals directly with the integration of heterogeneous contexts (i.e. data meaning) in a flexible, scalable and extensible environment. The COntext INterchange (COIN) System [6] makes it easier and more transparent for receivers (e.g., applications, sensors, users) to exploit distributed sources (e.g., databases, web, information repositories, sensors). Receivers are able to specify their desired context so that there will be no uncertainty in the interpretation of the information coming from heterogeneous sources. The approach and associated tools significantly reduces the overhead involved in the integration of multiple sources and simplifies maintenance in an environment of changing source and receiver context.

An overview diagram of this approach is shown in Figure 1. The COIN project provides for a systematic representation and automatic processing of data semantics. Instead of explicitly capturing semantic conflicts, the COIN approach records data semantics declaratively and uses a mediation engine to detect all conflicts, which are reconciled by rewriting user queries to incorporate conversions that can be defined either internally or remotely on the network. This approach provides great extensibility.

We refer readers to [2,3] for a formal description of the COIN approach. The COIN framework is built on a deductive object-oriented data model where semantic data types and their properties are represented in an ontology. A modifier is a kind of property that determines how an instantiated semantic object is interpreted in different contexts. Data semantics are declared with 1) elevation axioms that map data elements to the semantic types in the ontology; and 2) context definitions that specify modifier values.

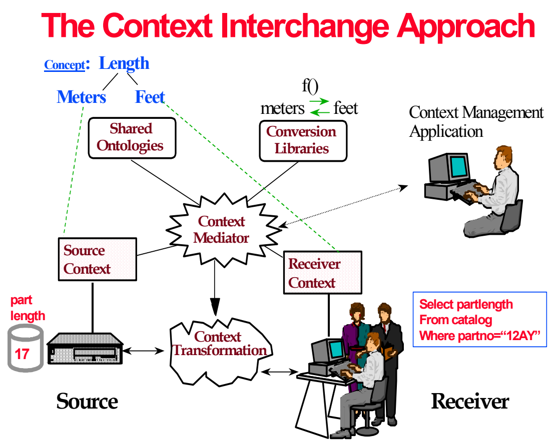

Figure 1. Context Interchange Approach

An abductive reasoning engine is used to detect semantic conflicts and rewrite the query into one that resolves the conflicts. COIN also implements tools for authoring ontologies, interfacing with other representation (e.g., RDF, OWL)[7], specifying contexts, merging applications and domains, optimizing and executing queries.

2. Demonstration

In what follows, we will describe a COIN demonstration using several prototype applications that perform meaningful comparison of data from web sources. We demonstrate COIN, COIN development tools, and COIN application merging capabilities.

2.1 Context Mediation in a Single Application

The first part of the demonstration presents a Flight Reservation System. This application makes use of databases and web pages for flight scheduling and cost information. The sources may use differing meanings for flight information. The demonstration uses SQL as the intermediate query language and MIT's Chameleon Web extraction tool [8] to access semi-structured information from web pages. The COIN abduction engine identifies semantic conflicts through the comparison of modifier values (i.e. declarative context). Resolution of conflicts takes place automatically. The application returns a set of flight information in the context desired by the user.

The system accepts user queries in SQL, which are converted into Datalog by the query compiler. The user query is expressed as if all sources were in the user's context. The mediation engine then generates a mediated query that reconciles semantic differences, if any, between all sources involved and the user. The query optimizer and executioner [1] implements a capability aware and cost based distributed query optimization algorithm that takes advantage of parallel execution of subqueries in multiple sources.

We show the underlying components of the system including the ontology, contexts (i.e. modifiers) and elevation axioms. We then demonstrate the abduction and automatic rewriting capabilities of the system. This part of the demonstration also includes a look at the SQL interface to the set of sources of the application.

The specific application involves the aggregation of a number of web travel sources. The ontology for the Flight Reservation System is shown in Figure 2. In the ontology, semantic data types are shown in rounded boxes. Every type will be represented using primitive data types that are collectively called the *basic* type. Attributes and modifiers have these types as their domains and ranges. A context is a specification of all modifiers in the ontology. For instance, modifier *currency* may be specified to be "USD" for a U.S. context. What is included in *price* (e.g., taxes included or not) can be an ontological problem. But it is more flexible to model it as a context problem where each context is specified in modifier *type* and conversions between contexts are solved using the symbolic equation solver implemented in the mediation engine [9].

In Figure 3 we present the context issues for this demonstration. Here the user context are represented by Dora. Dora, for example, would like to see airfares that are in US dollars and that include paper ticket charges and service fees. The sources have context values (i.e., modifier values) different from Dora's. For example, TravelSelect provides airfares in British Pounds and does not include paper ticket charges or services fees.

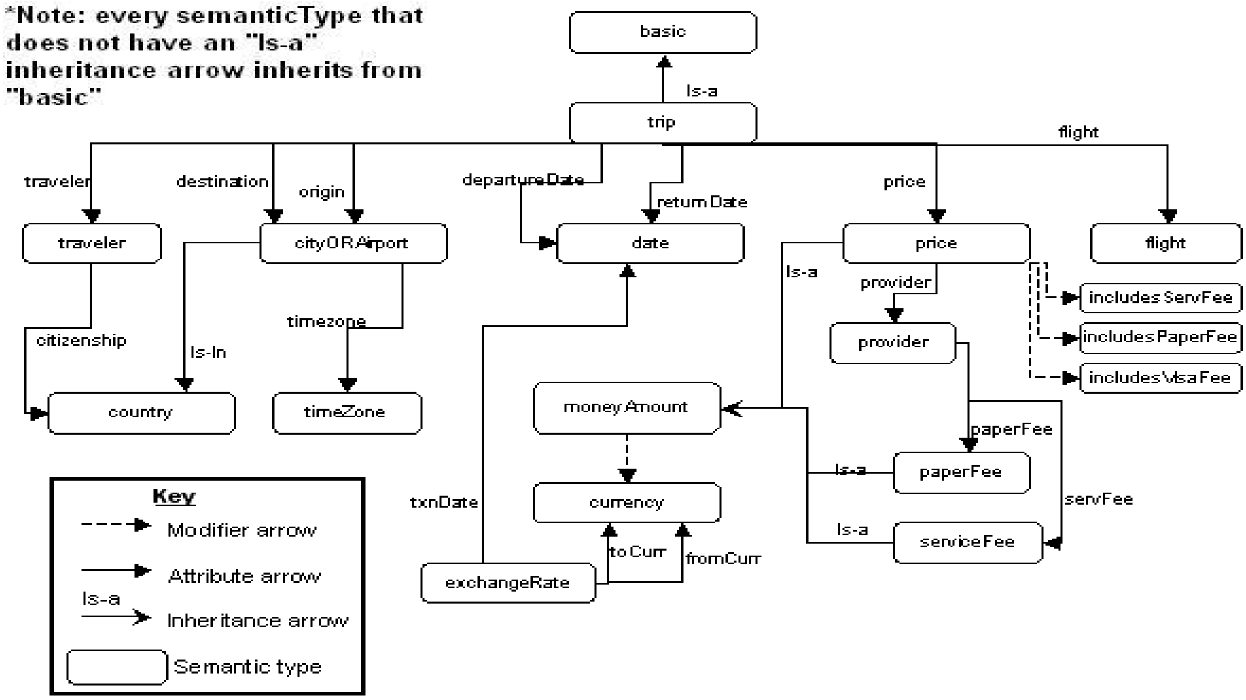

Figure 2. The Flight Reservation Ontology

Context Type	Context Name	includes-ServiceFee	includesPa-perTktCharge	currency
Receiver Contexts	Dora's Friend	No	No	GBP
	Dora	Yes	Yes	USD
Source Contexts	Yahoo	Yes	No	USD
	Expedia	Yes	No	USD
	Orbitz	No	No	USD
	Travelselect	No	No	GBP
	Itn	No	No	USD

Figure 3. Context Values for Sources and Receivers

Once the mediator has determined a conflict, conversion functions are used to provide the results in the users context. For currency conversion, we also need an internal source to generate the current date and an online source, *Olsen* at oanda.com, for exchange rate. Conversion functions for translating between the data level (*e.g.*, currency differences) and the ontological level (*e.g.*, what is included in price) are defined using rules.

Assume that Dora requests information from Travelselect and Yahoo. The conflicts are identified and the conversions, as shown in Figure 4 are executed automatically. The results are provided to Dora in her context.

Source	Conversion
Yahoo	No need to add taxes, service fees already included (do nothing), determine paper ticket charge and add it.
Travelselect	No need to add taxes, determine service fee and paper ticket charge for Orbitz and add them, convert everything from GBP to USD.

Figure 4. Conversion from Source Context to Dora's Context

A second application, for Car Rental, was developed and will be shown so that we can demonstrate the merging capabilities of COIN.

2.2 COIN Authoring Tools

We use a set of web-based authoring tools [4] to create and manage the ontology, the elevation axioms, and context definitions, which we call the knowledgebase for the application. This tool also imports RDF and exports RDF

[5,7]. By this means we can utilize ontologies developed by other applications. The tool provides both a text-based and a graphical interface. Using this tool we will demonstrate the ability to develop context knowledge, to add a new source and to modify context.

2.3 Merging Application Domains: A Travel Agent

Applications are developed in particular domains of interest. These domains are managed and used by the application using domain models (i.e., ontologies and context). It is important that the effort to develop these applications and associated domain models be reusable in other applications that may draw from one or more application domains. For this purpose we have developed technology for application domain merging. Unlike other approaches we utilize existing domain models intact. We have developed a tool that creates merging axioms that reside with the new application and operate over existing ontologies and contexts.

In the last part of the demonstration we will show the merging process. We will demonstrate a merged application (i.e., Travel Agency which includes Flight Reservation and Car Rental). This will show how new applications can be developed using existing applications covering multiple domains. The merged application provides a number of new capabilities by using the context representations and the conversion functions of the underlying applications. For example, currency as a context value is included in the airfare application but not the car rental application, in the merged application one can rent cars from agencies that price in currencies other than US dollars.

3. Summary

In this demonstration we show the capabilities of COIN for context mediation and application merging. We include a demonstration of the new symbolic equation solving [9] and multi-application merging capabilities. COIN can be used to solve a spectrum of data semantics problems, including representational, ontological and temporal semantics. We have demonstrated these capabilities in a number of application domains, such as financial services [9], online shopping [12], disaster relief efforts [4], corporate house holding knowledge engineering [11], and larger applications built by combining existing ones (*e.g.*, combine an airfare aggregator and a car rental shopper into a travel planner, see demos at our website). Efforts are also underway to use COIN framework as a cost effective alternative to standardization in the financial industry [10]. In addition, we have developed a .NET version of web wrapper and performed a preliminary study on accessing data and methods using Web Services. Progress in these areas will make COIN technology available to the Semantic Web community. Other planned extensions, such as temporal context, will further improve the applicability of COIN technology for various data integration needs.

References

1. Alatovic, T. (2002) "Capabilities Aware Planner/Optimizer/Executioner for Context Interchange Project", MIT EECS Master's Thesis.
2. Goh, C.H. (1997) Representing and Reasoning about Semantic Conflicts in Heterogeneous Information Systems, PhD Thesis, MIT Sloan School of Management
3. Goh, C.H., Bressan, S., Madnick, S., Siegel, S. (1999) "Context Interchange: New Features and Formalisms for the Intelligent Integration of Information", *ACM Transactions on Information Systems,* **17**(3), 270-293
4. Lee, P.W. (2003) Metadata Representation and Management for Context Mediation, MIT EECS Master's Thesis.
5. Liburd, S. (2003) Conceptual Mapping and Structural Conversions between COIN and

RDF, technical report, MIT Sloan School of Management.

6. Madnick, S.E. (1999) "Metadata Jones and the Tower of Babel: The Challenge of Large-Scale Semantic Heterogeneity", *Proceedings of the 1999 IEEE Meta-Data Conference*.
7. Manola, F. Miller, E. (2003) "RDF Primer", W3C working draft.
8. Firat, A., Madnick, S., Siegel, M. (2000) "The Cameleon Web Wrapper Engine", *Proceedings of the Workshop on Technologies for E-Services*, September 14-15, Cairo, Egypt.
9. Firat, A., Madnick, S., Grosof, B. (2002) Financial Information Integration in the Presence of Equational Ontological Conflicts, *12th WITS*, December 14-15, Barcelona, Spain.
10. Moulton A., Siegel, M.D., Madnick, S.E. (2003) Potential for Cost Savings from Application of Context Mediation Technology to Problems of Standards Interoperability in the Financial Services Industry, internal report, MIT Sloan School of Management.
11. Xian, X. (2003) Corporate Householding Knowledge Engineering and Processing using Extended COIN, MIT EECS Master's Thesis.
12. Zhu, H. Madnick, S., Siegel, M. (2002) "Global Comparison Aggregation Services", *1st Workshop on E-Business*, December 14-15, Barcelona, Spain.