Pre-print from IEEE Data Engineering, June 1990.

THE COMPOSITE INFORMATION SYSTEMS LABORATORY (CISL) PROJECT AT MIT

Stuart E. Madnick, Michael Siegel, Y. Richard Wang

May 1990 WP# 3157-90
E53-321, Sloan School of Management
Massachusetts Institute of Technology
Cambridge, MA 02139

Pre-print from IEEE Data Engineering, June 1990.

The Composite Information Systems Laboratory (CISL) Project at MIT

Stuart E. Madnick
John Norris Maguire Professor
Sloan School of Management
Massachusetts Institute
of Technology
E53-321
Cambridge, MA 02139
(617) 253-6671
smadnick@sloan.mit.edu

Michael Siegel
Research Associate
Sloan School of Management
Massachusetts Institute
of Technology
E53-323
Cambridge, MA 02139
(617) 253-2937
msiegel@sloan.mit.edu

Y. Richard Wang
Assistant Professor
Sloan School of Management
Massachusetts Institute
of Technology
E53-317
Cambridge, MA 02139
(617) 253-0442
rwang@sloan.mit.edu

ABSTRACT

The Composite Information Systems Laboratory (CISL) at MIT is involved in research on the strategic, organizational and technical aspects of the integration of multiple heterogeneous database systems. In this paper we examine the scope of the work being done at CISL. Certain research efforts in the technical areas of system integration are emphasized; in particular semantic aspects of the integration process and methods for tracking data sources in composite information systems.

INTRODUCTION

The increasingly complex and globalized economy has driven many corporations to expand business beyond their traditional organizational and geographic boundaries. It is widely recognized today that many important applications require access to and integration of multiple heterogeneous database systems. We have referred to these types of application systems as *Composite Information Systems* (CIS).

A fundamental CIS assumption is that organizations must deal with and connect pre-existing information systems which have been developed and administered independently. With this assumption, CIS follows two principles: (1) *system non-intrusiveness*, and (2) *data non-intrusiveness*. By *system non-intrusiveness* we mean that a pre-existing information system need not be changed. We have found that many of these systems are controlled by autonomous organizations or even separate corporations that are reluctant or unwilling to change their systems. By *data non-intrusiveness* we mean that changes are not required to the data in a pre-existing information system. Although we are in favor of data standardization, we have found that it is difficult to attain in a timely manner across organizational boundaries.

SCOPE OF CISL PROJECT

The CISL project is a broad-based research undertaking with nine component efforts as depicted in Figure 1. The remainder of this paper will focus on the technical aspects of prototype implementation and theory development as depicted in cell 8 and 9 of Figure 1.

Acknowledgements: Work reported herein has been supported, in part, by Citibank, IBM, Reuters, MIT's International Financial Service Research Center (IFSRC), MIT's Laboratory for Computer Science (LCS), and MIT's Leaders for Manufacturing (LFM) program.

Type of Connectivity	Methodologies: Field Studies	Prototype Implementation	Theory Development
Strategic	1) Medium	2) Low	3) Medium
Organizational	4) Medium	5) Low	6) Medium
Technical	7) High	8) **High**	9) **High**

Figure 1. Scope and Intensity of CISL Research Activities

A key objective of a CIS is to provide connectivity among disparate information systems that are both internal and external to an organization. Three types of connectivity have been researched:

1. Strategic Connectivity: The identification of the strategic requirements for easier, more efficient, integrated intra-organizational and inter-organizational access to information [MA88].

2. Organizational Connectivity: The ability to connect interdependent components of a loosely-coupled organization in the face of the opposing forces of centralization (e.g., in support of strategic connectivity) and decentralization (e.g., in response to the needs of local conditions, flexibility, distribution of risk, group empowerment) [OS89].

3. Technical Connectivity: The technologies that can help a loosely-coupled organization appear to be more tightly-coupled. This area is the primary focus of this paper and will be elaborated upon below.

The CISL research has employed three methodologies:

1. Field Studies: We have conducted detailed studies of several major corporations and government agencies to understand their current connectivity situation, future requirements and plans, and major problems encountered or anticipated [GA89, GO89, PA89, WA88].

2. Prototype Implementation: A prototype CIS, called the Composite Information Systems/Tool Kit (CIS/TK), has been developed to test solutions to many of the problems identified from the field studies [W089].

3. Theory Development: In several areas problems have been found for which no directly relevant existing theories have been identified [SI89a, SI89b, WA89c, WA90]. Two particular technical connectivity theory development efforts described in this paper are *semantic reconciliation* which deals with the integration of data semantics among disparate information systems and *source tagging* which keeps track of originating and intermediate data sources used in processing a query.

CURRENT CIS PROTOTYPE STATUS

Our current CIS prototype, CIS/TK Version 3.0, has been demonstrated to provide access to as many as six disparate databases. The three MIT databases use different dialects of SQL and are run by different MIT organizations: **alumni** database (Informix-SQL on an AT&T 3B2 computer), **recruiting** database (oracle-SQL on an IBM RT), and **student** database (SQL/DS on an IBM 4381). Finsbury's **Dataline** service appears hierarchical to the end-user and has a menu-driven query interface. I.P. Sharp's **Disclosure** and **Currency** services provide both proprietary menu and query language interfaces. These databases provide breadth in data and provide examples of differences in style, accentuated somewhat by the different origins of each service (e.g., Finsbury is based in London, England and I.P. Sharp is based in Toronto, Canada).

CIS/TK Version 3.0 was completed in August 1989. It runs on an AT&T 3B2/500 computer under UNIX System V. Most of it is implemented in our object-oriented rule-based extension of Common Lisp, called the Knowledge-Oriented Representation Language (KOREL). Certain components are implemented in C and UNIX Shell. The current system provides direct access to, and integration of, information from all six databases.

We now present a simplified version of the actual heterogeneous CIS environments of CIS/TK. Suppose, as part of a major new marketing campaign, we wanted to know our top hundred customers in terms of total purchases from our two divisions over the past five years, expressed in dollars. Total purchases for a customer is calculated by summing purchases from each of the two divisions. In order to *reconcile the semantic heterogeneities*, two major tasks need to be addressed: (a) inter-database instance matching and (b) value interpretation and coercion [WA89a, WA89b].

Instance Matching

Each division may identify a customer differently. Thus, it will be necessary to match a customer in one database to the same customer in the other by means other than an exact string comparison. For example, "IBM Corp." in one should be matched to "IBM, Inc." in the other; also "MIT" should match "Mass. Inst. Tech."; and "Continental Airlines" should match "Texas Air Corp.". Each division may also use a customer number (or other unique identifier) for a customer, but it is unlikely that the same identifier would have been used by both for a given customer.

To match customers (or any entity) between two databases, we use a combination of three techniques: key semantic matching, attribute semantic matching, and organizational affinity.

- *Key Semantic Matching*: To match the two IBMs, we can use rules such as "Corp." and "Inc." suffixes are equivalent. These rules can be context sensitive.

- *Attribute Semantic Matching:* The two MIT's are harder cases because it is unlikely that we would have a rule that "M" and "Mass." were equivalent. Instead we identify the match based on the values of other attributes (such as both have CEO "Paul Grey" and HQ city "Cambridge").

- *Organizational Affinity*: In many cases there exists an affinity between two distinct organizations. For example, although "Continental Airlines" is a separate corporation from "Texas Air Corp", Texas Air Corp owns Continental. Thus, depending on the purpose of the query, it may be desirable to treat two distinct entities as being the same (e.g., from a marketing perspective Continental and Texas Air may be merged; from a legal liability perspective they should be kept separate).

Value Interpretation and Coercion

After instance matching is performed, we must compute the combined sales to each customer. This presents numerous challenges regarding value interpretation and value coercion.

- *Value Interpretation:* For example, in one division, the total purchase amount is expressed as a character string that combines the currency indicator and numeric value (e.g., ¥120,000). The other division does not explicitly identify the currency being used; it must be inferred from the country of customer information (USA = $, Japan = ¥).

- *Value Coercion:* In order to compute the total purchase in dollars (or to compare totals across customers), a currency exchange rate needs to be applied. Since the yen/dollar exchange rate varies considerably from year to year (if not moment to moment), it is also necessary to "know" what time period to use and where to find the currency exchange data for that time period.

Although many of the instance matching, value interpretation and coercion features described above are implemented in a rudimentary form in CIS/TK Version 3.0, it still provides an excellent tool for testing new algorithms and approaches. In the remainder of this paper we examine our present research efforts in the development of new algorithms and approaches for composite information systems.

SEMANTIC RECONCILIATION USING METADATA (Source-Receiver Problem)

It has become increasingly important that methods be developed that explicitly consider the meaning of data used in information systems. For example, it is important that an application requiring financial data in francs does not receive data from a source that reports in another currency. This problem is a serious concern for many corporations because the source meaning may change at any time; a source that once supplied financial data in francs might decide to change to reporting that data in European Currency Units (ECUs).

To deal with this problem, the system must be able to represent data semantics and detect and automatically resolve conflicts in data semantics. At best, present systems permit an application to examine the data type definitions in the database schema, thus allowing for type checking within the application. But this limited capability does not allow a system to represent and examine detailed data semantics nor handle changing semantics.

We have examined the specification and use of metadata in a simple *source-receiver* model [SI89a, SI89b]. The *source* (database) supplies data used by the *receiver* (application). Using metadata we described a method for determining semantic reconciliation between a source and a receiver (i.e., whether the semantics of the data provided by the source is meaningful to the receiver).

The need to represent and manipulate data semantics or metadata is particularly important in composite information systems where data is taken from multiple disparate sources. Typically, schema integration algorithms have been developed for component databases with static structure and semantics. However, to allow for greater local database autonomy, schema integration must be considered a dynamic problem. The global schema must be able to evolve to reflect changes in the structure and meaning of the underlying databases. If an application is affected by these changes, it must be alerted. As part of our research we are developing methods that use metadata to simplify schema integration while allowing for greater local database autonomy in an evolving heterogeneous database environment.

Methods for semantic reconciliation are described within a well-defined model for data semantics representation. This representation assumes that there are common primitive data types. From this base, for example, the currency of a trade price can be defined as the currency of the exchange where it was traded. Using this common language, sources can define the semantics of the data they supply and applications, using the same language, can define the semantic specification for required data.

The system architecture for semantic reconciliation is shown in Figure 2. Rather than a direct connection between the application and the database, the system includes a Database Metadata Dictionary component which contains knowledge about the semantics of the data and an Application Metadata Specification component which contains knowledge about the semantic requirements of the application. Semantic reconciliation is needed to determine if the data supplied by the database meets the semantic requirements of the application.

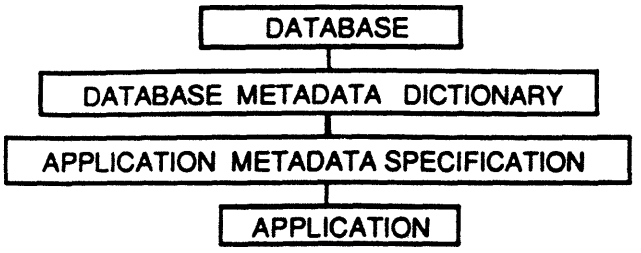

Figure 2. Semantic Reconciliation - System Architecture

We have developed an algorithm that compares the semantic requirements of the application with the meaning of the data supplied by the source to determine if the source will supply meaningful data [SI89b]. A similar algorithm can be used to determine if an application is receiving meaningful data from a set of component databases. In this case the data semantics of each database are examined to determine which sources might provide meaningful data. These methods can also be used to determine if an application can continue in the presence of changes in the component database semantics by making use of available conversion routines. Thus semantic reconciliation is a dynamic process which allows for component database semantic autonomy.

POLYGEN MODEL (Source Tagging Problem)

A typical objective of a distributed heterogeneous DBMS is that users must be able to access data without knowing where the data is located. In our field studies of actual needs, we have found that although the users want the simplicity of making a query as if it were a single large database, they also want the ability to know the source of each piece of data retrieved.

A polygen model has been developed to study heterogeneous database systems from this multiple (*poly*) source (*gen*) perspective [WA89c, WA90]. It aims at addressing issues such as "where is the data from," "which intermediate data sources were used to arrive at that data," and "how source tags can be used for information composition and access charge purposes." In a complex environment with hundreds of databases, all of these issues are critical to their effective use.

The polygen model developed presents a precise characterization of the source tagging problem and a solution including a polygen algebra, a data-driven query translation mechanism, and the necessary and sufficient condition for source tagging. The polygen model is a direct extension of the relational model to the multiple database setting with source tagging capabilities, thus it enjoys all of the strengths of the traditional relational model. Knowing the data source enables us to interpret the data semantics more accurately, knowing the data source credibility enables us to resolve potential conflicts amongst the data retrieved from different sources, and knowing the cost of accessing a local database enables us to develop an access charge system.

A polygen domain is defined as a set of ordered triplets. Each triplet consists of three elements: a datum drawn from a simple domain in a local database (LD), a set of LDs denoting the local databases from which the datum originates, and a set of LDs denoting the intermediate local databases whose data led to the selection of the datum. A polygen relation p of degree n is a finite set of time-varying n-tuples, each n-tuple having the same set of attributes drawing values from the corresponding polygen domains.

We have developed precise definitions of a polygen algebra based on six orthogonal operators: project, cartesian product, restrict, union, difference, and coalesce. The first five are extensions of the traditional relational algebra operators to operate on the data tags, whereas coalesce is a special operator needed to support the polygen algebra by merging the data tags from two columns into a single column. Other important operators needed to process a polygen query can be defined in terms of these six

operators, such as outer natural primary join, outer natural total join, and merge. A query processing algorithm to implement a polygen algebra has also been developed.

CONCLUDING REMARKS

This article has presented a brief overview of the CISL project objectives with specific focus on the prototype implementation and theory development regarding semantic reconciliation and source tagging. All of the research activities depicted in Figure 1 are currently underway and have been, or will be, reported in a series of CISL project reports and articles. From our interviews of major organizations, the problems being addressed are becoming increasingly critical to the development and deployment of modern information systems.

REFERENCES

[Due to limited space, only CISL project references are listed here. Citations to background work and other related projects can be found in the reference sections of these papers.]

[GO89] D.B. Godes, "Use of Heterogeneous Data Sources: Three Case Studies, "Sloan School of Management, MIT, Cambridge, MA. CISL Project, WP # CIS-89-02, June 1989.

[GU89] A. Gupta, S. Madnick, C. Poulsen, T. Wingfield, "An Architectural Comparison of Contemporary Approaches and Products for Integrating Heterogeneous Information Systems," Sloan School of Management, MIT, Cambridge, MA. WP # 3084-89 and IFSRC # 110-89, November 1989.

[MA88] S. Madnick and R. Wang, "Evolution Towards Strategic Applications of Data Bases Through Composite Information Systems," *Journal of MIS, Vol. 5*, No. 2, Fall 1988, pp.5-22.

[OS89] C. Osborne, S. Madnick and R. Wang, "Motivating Strategic Alliance for Composite Information Systems: The Case of a Major Regional Hospital," *Journal of MIS, Vol. 6*, No. 3, Winter 1989/90, pp.99-117.

[PA89] M.L. Paget, "A Knowledge-Based Approach Toward Integrating International On-line Databases," Sloan School of Management, MIT, Cambridge, MA. CISL Project, WP # CIS-89-01, March 1989.

[SI89a] M. Siegel and S. Madnick, "Schema Integration Using Metadata," Sloan School of Management, MIT, Cambridge, MA, WP # 3092-89 MS, October 1989 and *1989 NSF Workshop on Heterogeneous Databases*, December 1989.

[SI89b] M. Siegel and S. Madnick, "Identification and Reconciliation of Semantic Conflicts Using Metadata," Sloan School of Management, MIT, Cambridge, MA, WP # 3102-89 MSA, October 1989.

[WA88] R. Wang and S. Madnick, *Connectivity Among Information Systems*, Composite Information Systems (CIS) Project, Vol. 1, 141 pages, 1988.

[WA89a] R. Wang and S. Madnick, "Facilitating Connectivity in Composite Information Systems," *ACM Database*, Fall 1989.

[WA89b] R. Wang and S. Madnick, "The Inter-Database Instance Identification Problem in Integrating Autonomous Systems," *Proceedings of the Fifth International Conference on Data Engineering*, February 6-10, 1989.

[WA89c] R. Wang and S. Madnick, "A Polygen Data Model for Data Source Tagging in Composite Information Systems," Sloan School of Management, MIT, Cambridge, MA, WP # 3100-89 MSA, 1989.

[WA90] R. Wang and S. Madnick, "A Polygen Model for Heterogeneous Database Systems: The Source Tagging Perspective," *Proceedings of the 16th International Conference on Very Large Data Bases*, August, 1990.

[WO89] T.K. Wong, "Data Connectivity for the Composite Information System/Tool Kit," Sloan School of Management, MIT, Cambridge, MA. CISL Project, WP # CIS-89-03, June 1989.