Query Processing the Heterogeneous Information Sources using Ontology-based Approach

Ngamnij Arch-int[*]
Department of Mathematics, Faculty of Science, Chulalongkorn University, Bangkok, 10330, Thailand
ngamnij@kku.ac.th

Yuefeng Li
School of Software Engineering and Data Communications, Queensland University of Technology, Brisbane, Australia
y2.li@qut.edu.au

Paul Roe
School of Software Engineering and Data Communications, Queensland University of Technology, Brisbane, Australia
p.roe@qut.edu.au

Peraphon Sophatsathit
Department of Mathematics, Faculty of Science, Chulalongkorn University, Bangkok, 10330, Thailand
Peraphon.S@chula.ac.th

Abstract

The problems of accessing and integrating heterogeneous information sources are becoming center-stage problems. One problem arising from accessing heterogeneous sources is semantic heterogeneity. In this paper, we propose a metadata dictionary based on domain ontology as an assistant mechanism for query processing the heterogeneous sources and resolving semantic heterogeneity. An XML-based data model is employed to manipulate and express the metadata dictionary contents. The inherent flexibility of XML technology enables system-wide interoperability suitable for a Web-based operations.

Keywords: Heterogeneous Information Sources, Domain Ontology, Query Processing.

1 INTRODUCTION

Recently, one of the most problems arising from accessing and integrating **h**eterogeneous **i**nformation **s**ources (hereafter HIS) is *semantic heterogeneity*. Such a problem occurs when there is a disagreement about the meaning, interpretation, or intended use of the same or related data [16]. Examples of such semantic heterogeneity problems are naming conflicts, data type conflicts, scaling conflicts, and generalization conflicts.

A number of systems have been proposed to cope with semantic heterogeneity problems. For example, mediator-based systems [19, 9] provide the inter-schema architecture for integrating access to data from different sources and converting data and queries into canonical formats via the mediator and wrapper components. Description logic-based systems [12, 2] offer a different approach to elaborate source description by means of description logic [4] for solving queries over multiple sources. Unlike the mediator approach, the description logic approach abstracts the heterogeneous sources from users through a global view. Content-descriptive metadata systems [11, 7] utilize annotation information that is tightly integrated with HTML as metadata to describe the contents of a web document. A survey and comparison of these systems can be found in [14].

In this paper, we propose a metadata dictionary extended from [1] as a means for resolving semantic heterogeneity and providing access and integration of HIS on the WWW. Access and retrieval of HIS focus on structured data sources, such as database systems, and semi-structured data sources, such as XML documents [18]. The metadata dictionary is designed based on domain ontology [10, 8, 17], which acts as a unifying framework for accessing and integrating data with different data models into a homogeneous logical user's view. In order to support system-wide interoperability suitable for a Web-based environment, we choose XML as a language for expressing the metadata dictionary contents, as well as providing flexibility and scalability in building and manipulating the ontology terminologies. These terminologies are subsequently shared by agents to access and retrieve real data from the underlying physical sources.

The remainder of the paper is structured as follows. Section 2 presents the metadata dictionary based on ontology modeling techniques. Section 3 presents the structuring of XML-based metadata dictionary. The XML-DTD obtained in the process is also illustrated. Section 4 illustrates the querying process in accessing and integrating HIS through the proposed metadata dictionary. Section 5 concludes the paper and suggests further research extension.

2 ONTOLOGY-BASED METADATA DICTIONARY

In this section, we focus on modeling the domain ontology, which is the fundamental building block of the metadata dictionary. The domain ontology has been modeled on the basis of a bottom-up design approach [5, 13]. The modeling process involves the schema translation of the underlying physical information sources into intermediate schemas via the E-R model [6], and schema integration of these intermediate schemas into a global conceptual schema in order to eliminate structural heterogeneity [3, 13]. Details of schema translation and integration can be found in [15].

[*] ***Address for Correspondence***: Department of Computer Science, Faculty of Science, Khon Kaen University, Khon Kaen, Thailand, 40002. Tel. and Fax. +66 43 342910.

[*] *Part of this work has been conducted while the author was a visiting academic at Queensland University of Technology, Brisbane, Australia*

Our approach focuses on extracting the ontology from the underlying global conceptual schema to obtain an explicit user-viewed representation. The ontology extraction is illustrated through the real university system and extracted into two levels of abstraction, namely, the conceptual level of abstraction and the physical level of abstraction, as follows.

2.1 The Conceptual Level Representation

The global conceptual schema is restructured into virtual schema, as illustrated in Figure 1, encompassing virtual concepts (or entities), virtual properties (or attributes), relationships, and construction rules. The virtual schema is an initial ontology represented by the Extended Entity-Relationship (EER) model. The virtual property `st_id` is an object identifier or key, `st_name`, and `st_salary` are ordinary properties whose values are atomic values, and `dept_id` is an object identifier reference or foreign key. To solve data type and scaling conflicts, the object identifier and ordinary properties can further designate additional domain properties represented by circles to specify a predefined type and scaling domains. For example, the domain properties of `st_salary` are of the predefined type "Float" and scaling domain "US$". As such, the same logical data items with different physical data types or unit types, such as "Double" and "AU$" from HIS, can be displayed in a uniform format. To solve generalization conflicts, an IS-A relationship is used as an arrow to connect a specific concept (e.g., `Instructor`) to a general concept (e.g., `Staff`).

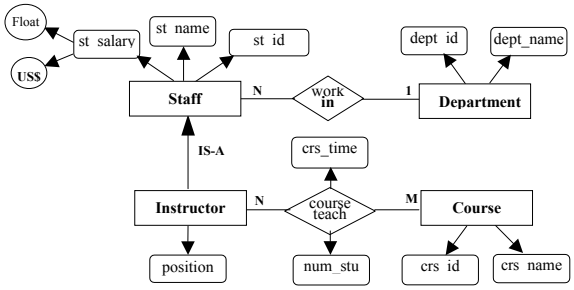


Figure 1. The ontology at the conceptual level of abstraction.

Note that the relationship `course_teach` is treated as a concept in the ontology.

2.2 The Physical Level Representation

This level is designed to solve naming conflicts by designing each virtual property to hold its physical instances (represented by ellipses) that store the synonymous physical property names of the physical concepts in a global conceptual schema. Figure 2 illustrates a partial ontology structure in this level. Each physical instance defines its own properties, denoted by circles that encompass other physical information corresponding to the physical instances, such as physical data type, unit type, concept, and source. For example, `Staff_id` of `Staff_Member` and `Inst_id` of `Instructor_Member` are synonymous terms and are designed as the physical instances of the virtual property `st_id`. The values of physical information properties named `PDataType`, `PUnitType`, `PCname`, and `PSname` of `Staff_id` are "Integer", "NULL", "Staff_Member", and "Source1", respectively. The ontology in this level also holds physical source configurations, furnishing necessary information to grant permission and knowledge for agents in accessing individual physical source.

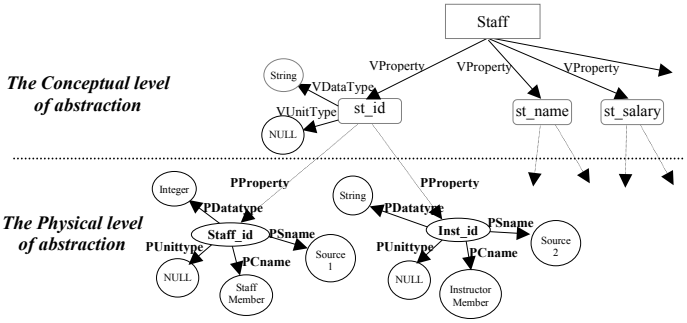


Figure 2. The ontology at the physical level of abstraction.

3 THE XML-BASED METADATA DICTIONARY

Since XML has strengths in well-formed, validity, and schema, we use XML to represent the metadata dictionary contents. The structural design of XML-DTD was set up from the domain ontology components to maintain their conceptual and physical correspondence and consistency as depicted in Figure 3.

```
<?xml version="1.0" standalone="yes"?>
<!DOCTYPE    MetadataDictionary [
  <!ELEMENT  MetadataDictionary  (VConcepts, PhysicalSourceConfs)>
  <!ATTLIST  MetadataDictionary  MetadataName  ID  #REQUIRED>
  <!ELEMENT  Vconcepts  (VConcept)+>
  <!ELEMENT  Vconcept   (VRelationships?, VProperties)>
  <!ATTLIST  Vconcept   Vcname  ID  #REQUIRED>
  <!ELEMENT  VRelationships   (VRelationship)+>
  <!ELEMENT  VRelationship    (AssocConcept)+>
  <!ATTLIST  VRelationship    VRelname        (IS-A|IS-PART-OF|Associative) #REQUIRED>
  <!ELEMENT  AssocConcept     (#PCDATA)>
  <!ATTLIST  AssocConcept     VConcept        IDREF          #IMPLIED>
  <!ELEMENT  VProperties      (VPoid|VPord|VPref)+>
  <!ELEMENT  VPoid            (VDataType, VUnitType, PProperties)>
  <!ATTLIST  VPoid            VPname          ID             #REQUIRED>
  <!ELEMENT  VPord            (VDataType, VUnitType, PProperties)>
  <!ATTLIST  VPord            VPname          CDATA          #IMPLIED>
  <!ELEMENT  VPref            (#PCDATA)>
  <!ATTLIST  VPref            VPoid           IDREF          #IMPLIED>
  <!ELEMENT  VDataType        (#PCDATA)>
  <!ELEMENT  VUnitType        (#PCDATA)>
  <!ELEMENT  PProperties      (PProperty)+>
  <!ELEMENT  PProperty        (PDataType, PUnitType)>
  <!ATTLIST  PProperty        PPname          CDATA          #REQUIRED
                              PCname          IDREFS         #REQUIRED
                              PSname          IDREF          #REQUIRED>
  <!ELEMENT  PDataType        (#PCDATA)>
  <!ELEMENT  PUnitType        (#PCDATA)>
  <!ELEMENT  PhysicalSourceConfs              (PSource)+>
  <!ELEMENT  PSource          (PConcept)+>
  <!ATTLIST  PSource          PSname                         ID             #REQUIRED>
  <!ELEMENT  PConcept         (PDataModel, Permission, Owner)>
  <!ATTLIST  PConcept         PCname                         ID             #REQUIRED>
  <!ELEMENT  PDataModel       (#PCDATA)>
  <!ELEMENT  Permission       (#PCDATA)>
  <!ELEMENT  Owner            (#PCDATA)>
]>
```

Figure 3. The XML-DTD metadata dictionary structure.

4 QUERYING PROCESSING FOR HETEROGENEOUS INFORMATION SOURCES

In the following, we discuss the querying process for HIS with the help of information from the metadata dictionary.

4.1 The Process of Accessing Heterogeneous Information Sources

The querying process to access HIS starts at the presentation layer of the reference architecture in [1]. Users can pose a query through a unified-query form encircling the virtual schema provided by the user interface agent. The processes in accessing HIS consist of two steps: global transaction creation and global transaction decomposition.

4.1.1 Global Transaction Creation

Upon submission of a user query, the request will be sent to the User Interface Agent to form a global transaction which is a visual user requirement represented in standard SQL format, as well as to validate the syntax by means of the metadata dictionary. The global transaction consists of virtual concepts and properties of the virtual schema. The global transaction is then sent to the managing agent, where global transaction decomposition is initiated.

4.1.2 Global Transaction Decomposition

This process transforms a global transaction into sub-transactions by substituting each virtual concept and property in the global transaction with the corresponding physical concept and property of the local physical sources obtained from the metadata dictionary. The decomposition processes can be accomplished in two steps as follows.

(1) ***Mapping.*** The virtual concepts and properties in the SELECT clause are mapped into the associated physical properties, concepts, and sources. This process is carried out through a mapping algorithm, as illustrated in Figure 4; and

(2) ***Sub-transactions creation.*** Each sub-transaction is created from the following processes:

2.1 ***Grouping process****:* The virtual concepts/properties and the corresponding physical concepts/ properties with the same physical source are grouped together.

2.2 ***Substitution process****:* The virtual concepts/properties in each group are substituted by the corresponding physical concepts/properties to form a sub-transaction. The physical properties are considered the requested information in the SELECT clause, and the physical concepts the target accessed information sources in the FROM clause.

Algorithm: Searching the Physical Information in the metadata dictionary.

Input: A pair <vc, vp>, where *vc* is a virtual concept name specified in the FROM clause and *vp* is a target virtual property name specified in the SELECT or WHERE clause

Output: $T = \{a_i \mid i = 1...n\}$, a finite set of virtual and physical mapping information such that $a_k \in T$ is defined as a 5-tuple < *vc, vp*, `PPname`, `PCname`, `PSname`>, where `PPname`, `PCname`, and `PSname` are physical property name, concept name and source name, respectively.

```
Let T = ϕ;
Let K = {"VConcepts", "VProperties", "PProperties"};
Let v = root;
for each unmark vertex w adjacent to v do
begin
        visit and mark w;
        if (w.element in K) or (w.VCname = vc) or
          (w.VPname = vp) or (w.VPref = vp) then
                v = w;
        else
                if v.element = "PProperties" then
                begin
                    Add (vc, vp, w.PPname, w.PCname, w.PSname) to T;
                end
end
```

Figure 4. The algorithm for searching physical information in the metadata dictionary.

2.3 ***Constraint generating process:*** The virtual concepts/ properties in the WHERE clause of the global transaction are mapped onto the associated physical properties, concepts, and sources. For each group with the same physical source, the qualifying predicates of the global transaction are replaced with the physical properties and concepts to form the same constraints in a sub-transaction. The join predicates of a sub-transaction are generated from matching the same pairs of physical properties and their corresponding physical concepts. These entire constraints are combined to construct the complete constraints of a sub-transaction as illustrated in Figure 5.

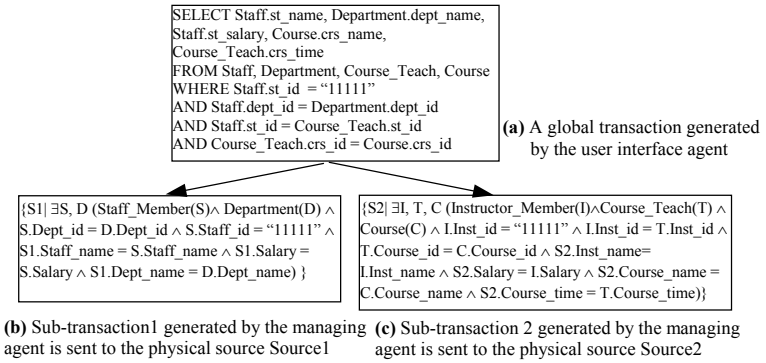


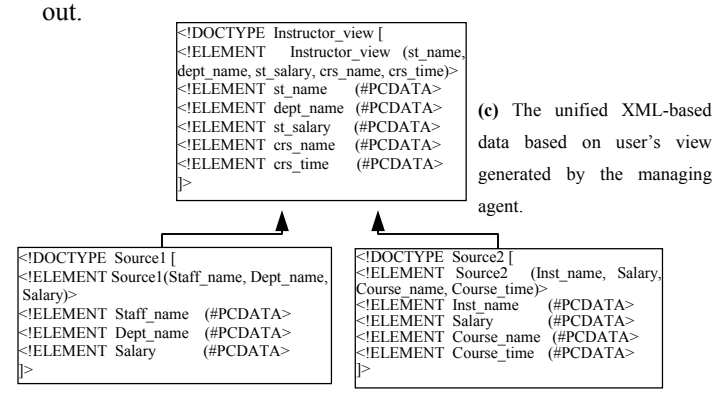
Figure 5. Decomposition of the global transaction into sub-transactions associated with physical information sources.

Each sub-transaction, together with the physical source configurations that are necessary for accessing HIS, is then packed and sent along with each search agent to the resource agent at the destination source.

4.2 The Process of Integrating Heterogeneous Information Sources

In this process, the results obtained from the execution of each sub-transaction are transformed into a canonical data model represented in an XML-based format via the interface wrappers as illustrated in Figure 6

(a) and (b). These XML results are transmitted to the managing agent where the integration process is carried out.



(c) The unified XML-based data based on user's view generated by the managing agent.

(a) The resulting XML-DTD generated from information source (Source 1) via wrapper

(b) The resulting XML-DTD generated from information source (Source 2) via wrapper

Figure 6. The integration of XML results into the unified XML-based data.

The managing agent utilizes information obtained from the metadata dictionary to integrate XML results into unified XML-based data. Naming, data type, and scaling conflicts are eliminated during this integration process as follows.

(1) ***Eliminating naming conflicts***: If any terms of each XML result are children of the same parent virtual property, these terms are treated as synonymous terms and are combined with the parent virtual property.

(2) ***Eliminating data type and scaling conflicts*****:** The different data types or unit types of the same elements or synonymous terms in each XML result are converted into a uniform data type or unit type defined in the associated virtual property.

The unified XML-based data generated from the corresponding conceptual virtual schema (as shown in Figure 6 (c)) is thus forwarded to the user interface agent, and eventually presented to the users at the presentation layer.

5 CONCLUSION

This work contributes to both theory and practice of working with HIS in many aspects. First, accessing and integrating HIS can be accomplished through the metadata dictionary approach. Second, the metadata dictionary provides a mapping mechanism to associate user's requests posed at the conceptual level with the physical level without loss of information in the query. Third, choosing XML technology to express the contents of the metadata dictionary renders maximal interoperability across heterogeneous systems. As such, metadata dictionary content management can be achieved by means of a flexible XML support. Our future work will focus on extending the metadata dictionary to support query planning and optimization.

6 REFERENCES

[1] N. Arch-int and P. Sophatsathit, "A Reference Architecture for Integrating Heterogeneous Information Sources using XML and Agent Model," Proc. of the 6th Joint Conference on Information Sciences, NC, USA, pp. 235-239, 2002.

[2] Y. Arens, C.Y. Chee, C.-N. Hsu and C. Knoblock, "Retrieving and Integrating Data from Multiple Information Sources," International Journal of Intelligent and Cooperative Information Systems Vol. 2(2), pp. 127-158, 1993.

[3] C. Batini and M. Lenzirini, "A Methodology for Data Schema Integration in Entity-Relationship Model," IEEE Transactions on Software Engineering, vol. 10(6), pp. 650-654, 1984.

[4] A. Borgida, "Description Logics in Data Management," IEEE Transactions on Knowledge and Data Engineering, vol. 7(5), pp. 671-682, 1995.

[5] S. Castano, V.D. Antonellis and S.D.C. Vimercati, "Global Viewing of Heterogeneous Data Sources," IEEE Transactions on Knowledge and Data Engineering, vol. 13(2), pp. 277-297, 2001.

[6] P.P. Chen, "The Entity-Relationship Model – Toward a Unified View of Data," ACM Transactions on Database Systems, vol. 1(1), pp. 9-36, 1976.

[7] S. Decker, M. Erdmann, D. Fensel and R. Studer, Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In: R. Meersman, et al. (eds.), Database Semantics: Semantic Issues in Multimedia Systems, Boston, MA, USA, pp. 351-369, 1999.

[8] D. Fensel, Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce, Springer-Verlag, Berlin, 2001.

[9] H. Garcia-molina, Y. Papakonstantinou, D. Quass, A. Rajaraman, Y. Sagiv, J. Ullman, V. Vassalos and J. Wisom, "The TSIMMIS approach to mediation: data models and languages," Journal of Intelligent Information Systems, vol. 8(2), pp. 117-132, 1997.

[10] T.R. Gruber, "A translation approach to portable ontology specifications," Knowledge Acquisition, vol. 4(2), pp. 199-220, 1993.

[11] V. Kashyap and A. Sheth, Semantic Heterogeneity in Global Information Systems: The Role of Metadata, Context and Ontologies. In: M. Papazoglou and G. Schlageter (eds.), Cooperative Information Systems: Current Trends and Directions, Academic Press London, 1998.

[12] A.Y. Levy, A. Rajaraman and J.J. Ordille, "Querying Heterogeneous Information Sources using Source Descriptions," Proc. of the 22nd International Conference on Very Large Databases, Bombay, India, pp. 251-262, 1996.

[13] M.T. Özsu and P. Valduriez, Principles of Distributed Database System, Second Edition, Prentice-Hall, New Jersey, 1999.

[14] N.W. Paton, C.A. Goble and S. Bechhofer, "Knowledge based information integration systems," International Journal of Information and Software Technology, vol. 42(5), pp. 299-312, 2000.

[15] M.P. Reddy, B.E. Prasad, P.G. Reddy and A. Gupta, "A Methodology for Integration of Heterogeneous Databases," IEEE Transactions on Knowledge and Data Engineering, vol. 6(6), pp. 920-933, 1994.

[16] A.P. Sheth and J.A. Larson, "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases," ACM Computing Surveys, vol. 22(3), pp. 183-236, 1990.

[17] M. Uschold and M. Gruninger, "ONTOLOGIES: Principles, Methods and Applications," The knowledge Engineering Review, vol. 11(2), pp. 93-155, 1996.

[18] W3C, World Wide Web Consortium, Extensible Markup Language (XML) 1.0 (Second Edition), http://www.w3.org/TR/2000/REC-xml-20001006, W3C Recommendation 6-Oct-2000.

[19] G. Wiederhold, "Mediators in the Architecture of Future Information Systems," IEEE computer, vol. 25(3), pp. 38-49,1992.