Martine Collard (Ed.)

LNCS 4623

Ontologies-Based Databases and Information Systems

First and Second VLDB Workshops, ODBIS 2005/2006
Trondheim, Norway, September 2005
Seoul, Korea, September 2006, Revised Papers

Springer

Martine Collard (Ed.)

Ontologies-Based Databases and Information Systems

First and Second VLDB Workshops,
ODBIS 2005/2006
Trondheim, Norway, September 2-3, 2005
Seoul, Korea, September 11, 2006
Revised Papers

Springer

Volume Editor

Martine Collard
Université de Nice - Sophia Antipolis
Laboratoire I3S
Les Algorithmes, 2000 route des Lucioles, 06903 Sophia Antipolis, France
E-mail: Martine.Collard@unice.fr

Library of Congress Control Number: 2007936205

CR Subject Classification (1998): H.2.1, H.2.4, H.3

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743
ISBN-10 3-540-75473-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-75473-2 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

springer.com

© Springer-Verlag Berlin Heidelberg 2007
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12162703 06/3180 5 4 3 2 1 0

Lecture Notes in Computer Science 4623

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison
Lancaster University, UK

Takeo Kanade
Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler
University of Surrey, Guildford, UK

Jon M. Kleinberg
Cornell University, Ithaca, NY, USA

Friedemann Mattern
ETH Zurich, Switzerland

John C. Mitchell
Stanford University, CA, USA

Moni Naor
Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz
University of Bern, Switzerland

C. Pandu Rangan
Indian Institute of Technology, Madras, India

Bernhard Steffen
University of Dortmund, Germany

Madhu Sudan
Massachusetts Institute of Technology, MA, USA

Demetri Terzopoulos
University of California, Los Angeles, CA, USA

Doug Tygar
University of California, Berkeley, CA, USA

Moshe Y. Vardi
Rice University, Houston, TX, USA

Gerhard Weikum
Max-Planck Institute of Computer Science, Saarbruecken, Germany

Preface

This volume constitutes the joint post-proceedings of the two international VLDB workshops on Ontologies-based Techniques for DataBases and Information Systems, ODBIS 2005 and ODBIS 2006, co-located with the 31st and 32nd International Conference on Very Large Data Bases (VLDB). It is a collection of extended versions of papers presented at the workshops.

Ontologies are generally used to specify and communicate domain knowledge in a generic way. While in a formal sense "ontology" means study of concepts, one can use the word "ontology" as a concept repository about a particular area of interest. Ontologies are very useful for structuring and defining the meaning of the metadata terms that are currently collected inside a domain community. They are a popular research topic in knowledge engineering, natural language processing, intelligent information integration and multi-agent systems. Ontologies are also applied in the World Wide Web community where they provide the conceptual underpinning for making the semantics of a metadata machine understandable. More generally, ontologies are critical for applications which want to merge information from diverse sources. They become a major conceptual backbone for a broad spectrum of activities dealing with databases and information systems. In these workshops, the objectives were to present databases and information systems research as they relate to ontologies and, more broadly, to gain insight into ontologies as they relate to databases and information systems. These post-proceedings are divided roughly into three sections: ontology-based interoperability and schema matching, management of ontological bases and links between ontologies and knowledge.

May 2007 Martine Collard

此为试读,需要完整PDF请访问: www.ertongbook.com

Editorial Board

ODBIS Co-chairs

Martine Collard	University of Nice-Sophia Antipolis, France
Jean-Louis Cavarero	University of Nice-Sophia Antipolis, France

Program Committee

Bill Andersen	Ontology Works, Baltimore, USA
Jurgen Angele	Ontoprise, Karlsruhe, Germany
Pascal Barbry	CNRS, Sophia Antipolis, France
Paolo Bouquet	University of Trento, Italy
Nieves R. Brisaboa	University of A Coruña, Spain
Marion G. Ceruti	Space and Naval Warfare Systems Center, San Diego, USA
Bruno Cremilleux	University of Caen, France
Monica Crubézy	Stanford Medical Informatics, USA
Isabel Cruz	University of Illinois, USA
Rose Dieng	INRIA, Sophia Antipolis, France
Peter W. Eklund	University of Wollongong, Australia
Maria-José Escalona	University of Seville, Spain
André Flory	University of Lyon, France
Carl-Chritian Kanne	University of Mannheim, Germany
Isabelle Mirbel	University of Nice-Sophia Antipolis, France
Michele Missikoff	Lab. for Enterprise Knowledge and Systems, IASI-CNR, Italy
Claire Nedellec	INRA, Jouy-en-Josas, France
Natasha Noy	Stanford Medical Informatics, USA
Nicolas Pasquier	University of Nice-Sophia Antipolis, France
Oscar Pastor	University of Valencia, Spain
Domenico Rosaci	University "Mediterranea" di Reggio Calabria, Italy
Heiner Stuckenschmidt	University of Amsterdam, The Netherlands
Gerd Stumme	University of Kassel, Germany
Vojtech Svtek	University of Economics, Prague, Czech Republic

Lecture Notes in Computer Science

Sublibrary 1: Theoretical Computer Science and General Issues

For information about Vols. 1–4475
please contact your bookseller or Springer

Vol. 4782: R. Perrott, B.M. Chapman, J. Subhlok, R.F. de Mello, L.T. Yang (Eds.), High Performance Computing and Communications. XIX, 823 pages. 2007.

Vol. 4770: V.G. Ganzha, E.W. Mayr, E.V. Vorozhtsov (Eds.), Computer Algebra in Scientific Computing. XIII, 460 pages. 2007.

Vol. 4763: J.-F. Raskin, P.S. Thiagarajan (Eds.), Formal Modeling and Analysis of Timed Systems. X, 369 pages. 2007.

Vol. 4746: A. Bondavalli, F.V. Brasileiro, S. Rajsbaum (Eds.), Dependable Computing. XV, 239 pages. 2007.

Vol. 4743: P. Thulasiraman, X. He, T.L. Xu, M.K. Denko, R.K. Thulasiram, L.T. Yang (Eds.), Frontiers of High Performance Computing and Networking ISPA 2007 Workshops. XXIX, 536 pages. 2007.

Vol. 4742: I. Stojmenovic, R.K. Thulasiram, L.T. Yang, W. Jia, M. Guo, R.F. de Mello (Eds.), Parallel and Distributed Processing and Applications. XX, 995 pages. 2007.

Vol. 4736: S. Winter, M. Duckham, L. Kulik, B. Kuipers (Eds.), Spatial Information Theory. XV, 455 pages. 2007.

Vol. 4732: K. Schneider, J. Brandt (Eds.), Theorem Proving in Higher Order Logics. IX, 401 pages. 2007.

Vol. 4731: A. Pelc (Ed.), Distributed Computing. XVI, 510 pages. 2007.

Vol. 4711: C.B. Jones, Z. Liu, J. Woodcock (Eds.), Theoretical Aspects of Computing – ICTAC 2007. XI, 483 pages. 2007.

Vol. 4710: C.W. George, Z. Liu, J. Woodcock (Eds.), Domain Modeling and the Duration Calculus. XI, 237 pages. 2007.

Vol. 4708: L. Kučera, A. Kučera (Eds.), Mathematical Foundations of Computer Science 2007. XVIII, 764 pages. 2007.

Vol. 4707: O. Gervasi, M.L. Gavrilova (Eds.), Computational Science and Its Applications – ICCSA 2007, Part III. XXIV, 1205 pages. 2007.

Vol. 4706: O. Gervasi, M.L. Gavrilova (Eds.), Computational Science and Its Applications – ICCSA 2007, Part II. XXIII, 1129 pages. 2007.

Vol. 4705: O. Gervasi, M.L. Gavrilova (Eds.), Computational Science and Its Applications – ICCSA 2007, Part I. XLIV, 1169 pages. 2007.

Vol. 4703: L. Caires, V.T. Vasconcelos (Eds.), CONCUR 2007 – Concurrency Theory. XIII, 507 pages. 2007.

Vol. 4700: C.B. Jones, Z. Liu, J. Woodcock (Eds.), Formal Methods and Hybrid Real-Time Systems. XVI, 539 pages. 2007.

Vol. 4698: L. Arge, M. Hoffmann, E. Welzl (Eds.), Algorithms – ESA 2007. XV, 769 pages. 2007.

Vol. 4697: L. Choi, Y. Paek, S. Cho (Eds.), Advances in Computer Systems Architecture. XIII, 400 pages. 2007.

Vol. 4688: K. Li, M. Fei, G.W. Irwin, S. Ma (Eds.), Bio-Inspired Computational Intelligence and Applications. XIX, 805 pages. 2007.

Vol. 4684: L. Kang, Y. Liu, S. Zeng (Eds.), Evolvable Systems: From Biology to Hardware. XIV, 446 pages. 2007.

Vol. 4683: L. Kang, Y. Liu, S. Zeng (Eds.), Advances in Computation and Intelligence. XVII, 663 pages. 2007.

Vol. 4681: D.-S. Huang, L. Heutte, M. Loog (Eds.), Advanced Intelligent Computing Theories and Applications. XXVI, 1379 pages. 2007.

Vol. 4672: K. Li, C. Jesshope, H. Jin, J.-L. Gaudiot (Eds.), Network and Parallel Computing. XVIII, 558 pages. 2007.

Vol. 4671: V. Malyshkin (Ed.), Parallel Computing Technologies. XIV, 635 pages. 2007.

Vol. 4669: J.M. de Sá, L.A. Alexandre, W. Duch, D. Mandic (Eds.), Artificial Neural Networks – ICANN 2007, Part II. XXXI, 990 pages. 2007.

Vol. 4668: J.M. de Sá, L.A. Alexandre, W. Duch, D. Mandic (Eds.), Artificial Neural Networks – ICANN 2007, Part I. XXXI, 978 pages. 2007.

Vol. 4666: M.E. Davies, C.J. James, S.A. Abdallah, M.D Plumbley (Eds.), Independent Component Analysis and Blind Signal Separation. XIX, 847 pages. 2007.

Vol. 4665: J. Hromkovič, R. Královič, M. Nunkesser, P. Widmayer (Eds.), Stochastic Algorithms: Foundations and Applications. X, 167 pages. 2007.

Vol. 4664: J. Durand-Lose, M. Margenstern (Eds.), Machines, Computations, and Universality. X, 325 pages. 2007.

Vol. 4649: V. Diekert, M.V. Volkov, A. Voronkov (Eds.), Computer Science – Theory and Applications. XIII, 420 pages. 2007.

Vol. 4647: R. Martin, M.A. Sabin, J.R. Winkler (Eds.), Mathematics of Surfaces XII. IX, 509 pages. 2007.

Vol. 4646: J. Duparc, T.A. Henzinger (Eds.), Computer Science Logic. XIV, 600 pages. 2007.

Vol. 4644: N. Azémard, L. Svensson (Eds.), Integrated Circuit and System Design. XIV, 583 pages. 2007.

Vol. 4641: A.-M. Kermarrec, L. Bougé, T. Priol (Eds.), Euro-Par 2007 Parallel Processing. XXVII, 974 pages. 2007.

Vol. 4639: E. Csuhaj-Varjú, Z. Ésik (Eds.), Fundamentals of Computation Theory. XIV, 508 pages. 2007.

Vol. 4638: T. Stützle, M. Birattari, H. H. Hoos (Eds.), Engineering Stochastic Local Search Algorithms. X, 223 pages. 2007.

Vol. 4628: L.N. de Castro, F.J. Von Zuben, H. Knidel (Eds.), Artificial Immune Systems. XII, 438 pages. 2007.

Vol. 4627: M. Charikar, K. Jansen, O. Reingold, J.D.P. Rolim (Eds.), Approximation, Randomization, and Combinatorial Optimization. XII, 626 pages. 2007.

Vol. 4624: T. Mossakowski, U. Montanari, M. Haveraaen (Eds.), Algebra and Coalgebra in Computer Science. XI, 463 pages. 2007.

Vol. 4623: M. Collard (Ed.), Ontologies-Based Databases and Information Systems. VII, 153 pages. 2007.

Vol. 4621: D. Wagner, R. Wattenhofer (Eds.), Algorithms for Sensor and Ad Hoc Networks. XIII, 415 pages. 2007.

Vol. 4619: F. Dehne, J.-R. Sack, N. Zeh (Eds.), Algorithms and Data Structures. XVI, 662 pages. 2007.

Vol. 4618: S.G. Akl, C.S. Calude, M.J. Dinneen, G. Rozenberg, H.T. Wareham (Eds.), Unconventional Computation. X, 243 pages. 2007.

Vol. 4616: A. Dress, Y. Xu, B. Zhu (Eds.), Combinatorial Optimization and Applications. XI, 390 pages. 2007.

Vol. 4614: B. Chen, M.S. Paterson, G. Zhang (Eds.), Combinatorics, Algorithms, Probabilistic and Experimental Methodologies. XII, 530 pages. 2007.

Vol. 4613: F.P. Preparata, Q. Fang (Eds.), Frontiers in Algorithmics. XI, 348 pages. 2007.

Vol. 4600: H. Comon-Lundh, C. Kirchner, H. Kirchner (Eds.), Rewriting, Computation and Proof. XVI, 273 pages. 2007.

Vol. 4599: S. Vassiliadis, M. Berekovic, T.D. Hämäläinen (Eds.), Embedded Computer Systems: Architectures, Modeling, and Simulation. XVIII, 466 pages. 2007.

Vol. 4598: G. Lin (Ed.), Computing and Combinatorics. XII, 570 pages. 2007.

Vol. 4596: L. Arge, C. Cachin, T. Jurdziński, A. Tarlecki (Eds.), Automata, Languages and Programming. XVII, 953 pages. 2007.

Vol. 4595: D. Bošnački, S. Edelkamp (Eds.), Model Checking Software. X, 285 pages. 2007.

Vol. 4590: W. Damm, H. Hermanns (Eds.), Computer Aided Verification. XV, 562 pages. 2007.

Vol. 4588: T. Harju, J. Karhumäki, A. Lepistö (Eds.), Developments in Language Theory. XI, 423 pages. 2007.

Vol. 4583: S.R. Della Rocca (Ed.), Typed Lambda Calculi and Applications. X, 397 pages. 2007.

Vol. 4580: B. Ma, K. Zhang (Eds.), Combinatorial Pattern Matching. XII, 366 pages. 2007.

Vol. 4576: D. Leivant, R. de Queiroz (Eds.), Logic, Language, Information and Computation. X, 363 pages. 2007.

Vol. 4547: C. Carlet, B. Sunar (Eds.), Arithmetic of Finite Fields. XI, 355 pages. 2007.

Vol. 4546: J. Kleijn, A. Yakovlev (Eds.), Petri Nets and Other Models of Concurrency – ICATPN 2007. XI, 515 pages. 2007.

Vol. 4545: H. Anai, K. Horimoto, T. Kutsia (Eds.), Algebraic Biology. XIII, 379 pages. 2007.

Vol. 4533: F. Baader (Ed.), Term Rewriting and Applications. XII, 419 pages. 2007.

Vol. 4528: J. Mira, J.R. Álvarez (Eds.), Nature Inspired Problem-Solving Methods in Knowledge Engineering, Part II. XXII, 650 pages. 2007.

Vol. 4527: J. Mira, J.R. Álvarez (Eds.), Bio-inspired Modeling of Cognitive Tasks, Part I. XXII, 630 pages. 2007.

Vol. 4525: C. Demetrescu (Ed.), Experimental Algorithms. XIII, 448 pages. 2007.

Vol. 4514: S.N. Artemov, A. Nerode (Eds.), Logical Foundations of Computer Science. XI, 513 pages. 2007.

Vol. 4513: M. Fischetti, D.P. Williamson (Eds.), Integer Programming and Combinatorial Optimization. IX, 500 pages. 2007.

Vol. 4510: P. Van Hentenryck, L.A. Wolsey (Eds.), Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems. X, 391 pages. 2007.

Vol. 4507: F. Sandoval, A.G. Prieto, J. Cabestany, M. Graña (Eds.), Computational and Ambient Intelligence. XXVI, 1167 pages. 2007.

Vol. 4502: T. Altenkirch, C. McBride (Eds.), Types for Proofs and Programs. VIII, 269 pages. 2007.

Vol. 4501: J. Marques-Silva, K.A. Sakallah (Eds.), Theory and Applications of Satisfiability Testing – SAT 2007. XI, 384 pages. 2007.

Vol. 4497: S.B. Cooper, B. Löwe, A. Sorbi (Eds.), Computation and Logic in the Real World. XVIII, 826 pages. 2007.

Vol. 4494: H. Jin, O.F. Rana, Y. Pan, V.K. Prasanna (Eds.), Algorithms and Architectures for Parallel Processing. XIV, 508 pages. 2007.

Vol. 4493: D. Liu, S. Fei, Z. Hou, H. Zhang, C. Sun (Eds.), Advances in Neural Networks – ISNN 2007, Part III. XXVI, 1215 pages. 2007.

Vol. 4492: D. Liu, S. Fei, Z. Hou, H. Zhang, C. Sun (Eds.), Advances in Neural Networks – ISNN 2007, Part II. XXVII, 1321 pages. 2007.

Vol. 4491: D. Liu, S. Fei, Z.-G. Hou, H. Zhang, C. Sun (Eds.), Advances in Neural Networks – ISNN 2007, Part I. LIV, 1365 pages. 2007.

Vol. 4490: Y. Shi, G.D. van Albada, J.J. Dongarra, P.M.A. Sloot (Eds.), Computational Science – ICCS 2007, Part IV. XXXVII, 1211 pages. 2007.

Vol. 4489: Y. Shi, G.D. van Albada, J.J. Dongarra, P.M.A. Sloot (Eds.), Computational Science – ICCS 2007, Part III. XXXVII, 1257 pages. 2007.

Vol. 4488: Y. Shi, G.D. van Albada, J.J. Dongarra, P.M.A. Sloot (Eds.), Computational Science – ICCS 2007, Part II. XXXV, 1251 pages. 2007.

Vol. 4487: Y. Shi, G.D. van Albada, J.J. Dongarra, P.M.A. Sloot (Eds.), Computational Science – ICCS 2007, Part I. LXXXI, 1275 pages. 2007.

Vol. 4484: J.-Y. Cai, S.B. Cooper, H. Zhu (Eds.), Theory and Applications of Models of Computation. XIII, 772 pages. 2007.

Table of Contents

A Multi-level Matching Algorithm for Combining Similarity Measures in Ontology Integration ... 1
Ahmed Alasoud, Volker Haarslev, and Nematollaah Shiri

Class Structures and Lexical Similarities of Class Names for Ontology Matching ... 18
Sumit Sen, Suman Somavarapu, and N.L Sarda

Scalable Interoperability Through the Use of COIN Lightweight Ontology ... 37
Hongwei Zhu and Stuart E. Madnick

Domain Ontologies Evolutions to Solve Semantic Conflicts ... 51
Guilaine Talens, Danielle Boulanger, and Magali Séguran

Requirements Ontology and Multi-representation Strategy for Database Schema Evolution ... 68
Hassina Bounif, Stefano Spaccapietra, and Rachel Pottinger

Improving the Development of Data Warehouses by Enriching Dimension Hierarchies with WordNet ... 85
Jose-Norberto Mazón, Juan Trujillo, Manuel Serrano, and Mario Piattini

Management of Large Spatial Ontology Bases ... 102
Evangelos Dellis and Georgios Paliouras

Knowledge Extraction Using a Conceptual Information System (ExCIS) ... 119
Laurent Brisson

The Semantic Desktop: A Semantic Personal Information Management System Based on RDF and Topic Maps ... 135
Edgar R. Weippl, Markus Klemen, Stefan Fenz, Andreas Ekelhart, and A Min Tjoa

Author Index ... 153

A Multi-level Matching Algorithm for Combining Similarity Measures in Ontology Integration

Ahmed Alasoud, Volker Haarslev, and Nematollaah Shiri

Computer Science & Software Engineering, Concordia University
1455 De Maisonneuve W., Montreal, Quebec, Canada
{ahmed_a,haarslev,shiri}@cse.concordia.ca

Abstract. Various similarity measures have been proposed for ontology integration to identify and suggest possible matches of components in a semi-automatic process. A (basic) Multi Match Algorithm (MMA) can be used to combine these measures effectively, thus making it easier for users in such applications to identify "ideal" matches found. We propose a multi-level extension of MMA, called MLMA, which assumes the collection of similarity measures are partitioned by the user, and that there is a partial order on the partitions, also defined by the user. We have developed a running prototype of the proposed multi level method and illustrate how our method yields improved match results compared to the basic MMA. While our objective in this study has been to develop tools and techniques to support the hybrid approach we introduced earlier for ontology integration, the ideas can be applied in information sharing and ontology integration applications.

1 Introduction

The rapid increase in the number of multiple information sources requires efficient and flexible frameworks for integration of these sources. Such frameworks should provide a way for extracting, transforming, and loading data from these sources, and be represented to the user in some appropriate way. There are two major approaches for integration of information: (1) the data warehouse (DW) or materialized approach and (2) virtual approach (also called mediator based).

In the context of ontology integration, we proposed a third approach [1] which is a hybrid between fully materialized and fully virtual approaches. Fig. 1 shows the architecture of this approach. The motivation of our ongoing research on integration of source ontologies was to develop tools and techniques for situations in which the information sources are expressed as ontologies, and to support queries over these sources, we need to build the global ontology (which has a common vocabulary among the sources). This allows the query processing (QP) component in the integrated framework in Fig. 1 to extract information from the ontology sources. To support this capability and realize the architecture proposed in Fig. 1, we need to develop effective matching techniques to assist users in a semi-automatic process. This is the motivation of the current work.

Let us review the issues faced in ontology matching, which is a fundamental problem in sharing information and integrating ontology sources in numerous

M. Collard (Ed.): ODBIS 2005/2006, LNCS 4623, pp. 1–17, 2007.
© Springer-Verlag Berlin Heidelberg 2007

applications. We witness a continuous growth in both the number and size of available ontologies developed to annotate knowledge on the web through semantics markups to facilitate sharing and reusing by machines. This, on the other hand, has resulted in an increased heterogeneity in the available information. For example, the same entity could be given different names in different ontologies or it could be modeled or described in different ways. The Ontology Matching Problem (OMP) may then be described as follows: given ontologies O1 and O2, each of which describing a collection of discrete entities such as classes, properties, individuals, etc., we want to find the semantic correspondences that exist between the components of these entities.

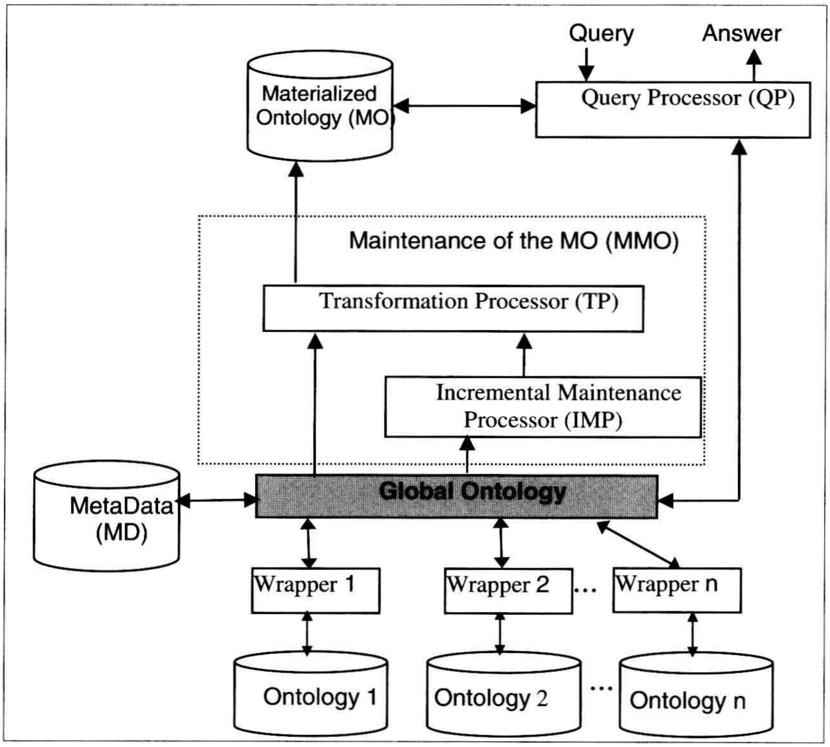

Fig. 1. The architecture of the hybrid framework [1]

Very often existing matching algorithms focus on one-to-one (1:1) matching. These methods hardly consider several entities at the same time and correspondingly use several similarity measures to solve OMP. In fact, OMP is an n:m matching problem. In order to obtain better matching results, existing measures should be used simultaneously and combined in a multi-space matching framework. We have developed such a method using a multi match algorithm (MMA).

此为试读，需要完整PDF请访问：www.ertongbook.com

The contributions of this paper are as follows:

1. We introduce an ontology matching approach, based on the idea of a multi-level match algorithm, in which each level uses different similarity measure(s).
2. We propose a flexible measure to compute the best possible matching state offered by MMA. This principle is based on the Dice coefficient adapted for our use.

The rest of the paper is organized as follows. In Section 2 we set up the formulation of the framework. The description of the algorithm is introduced in Section 3. An illustrative scenario is given in Section 4. The experiments and results are presented in Section 5. The related work is provided in Section 6. We conclude the paper with a summary and a discussion of future work in Section 7.

1.1 Motivating Example

In this section, we illustrate the ontology matching problem and introduce some concepts and techniques. Let us consider the following examples. Consider source ontology "S", which offers different types of electronic products. For simplicity, we consider only two products: PCs and laptops. Fig. 2 shows this ontology. As can be seen, S includes the concept *COMPUTERS* which represents *desktop* and *laptop*

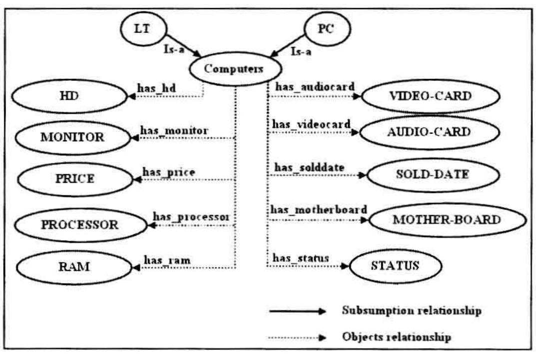

Fig. 2. Source ontology *S*

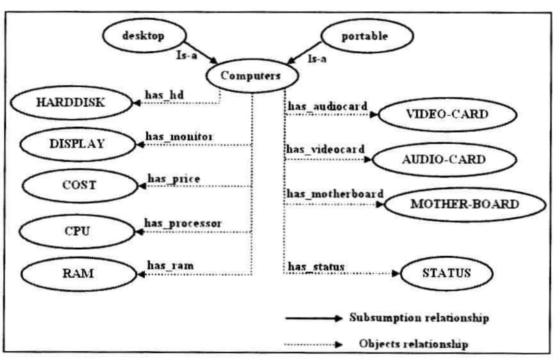

Fig. 3. Target ontology *T*

computers. Other concepts such as *MONITOR, PROCESSOR*, and *PRICE,* etc in this ontology represent technical specifications of computers. As the target ontology, we consider ontology "T", shown in Fig. 3. The goal is to find the corresponding matches among the entities in S and T.

There exist many methods to measure similarities between two entities, such as string similarity, linguistic similarity, etc. However, when we use a single matching measure for an input pair of ontologies, we may not be satisfied with the final match result. For instance, if we use a string similarity measure only, the concepts *PC* and *LT* in S have no matches in T. On the other hand, a string similarity measure is the basis for some other methods of measuring similarities between entities, and it works fine in some domains where a match in the entities on their syntax would most probably mean agreement on their semantics.

Another example is when we use a more semantic measure such as a linguistic based measure. For instance, we find out that the concept *PC* in S is mapped to the concept *desktop* in T and as well to concept *computer* in T. So, this will not help the user to focus his/her intention. As a result, if we use both measures (string and linguistic), the concept *computers* in S will be mapped into the concept *computers* in T with a very high confidence. Consequently, the concept *PC* in S will be mapped to *desktop* in T, and the concept *LT* in S will be mapped to *portable* in T.

1.2 General Description of the Framework

We propose a multi-level search algorithm that combines different measures in one unified framework to improve the matching results. Further, it minimizes user interaction with the system and suggests a single matching result of a collection of *n* elements in S to a collection of *m* elements in T.

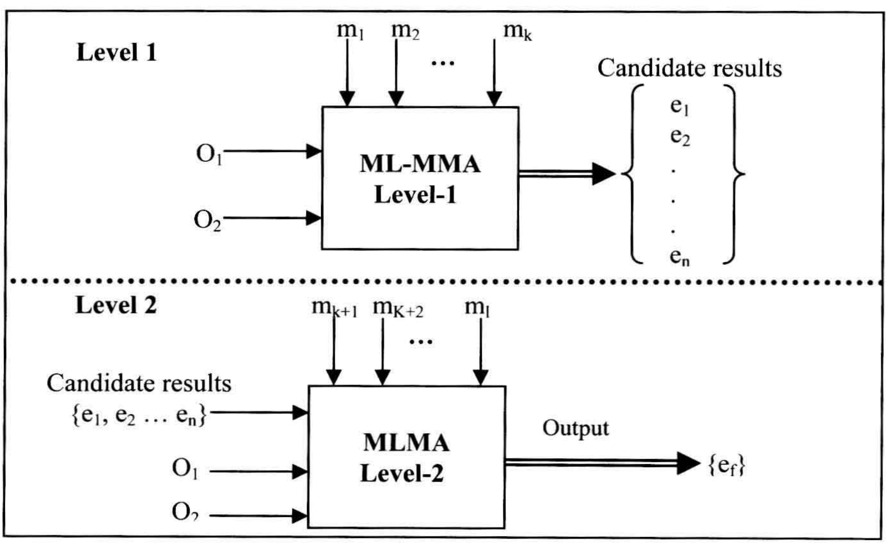

Fig. 4. A schematic description of the multi-level method

Fig. 4 illustrates the main idea of multi-level method, when there are two levels. It shows the different similarity measures {m_1, m_2,..., m_l} divided into two, and applied at two levels.

For instance, and to ease the presentation we use three similarity measures divided into two levels. The name and linguistic similarity measures have been applied in the first level. Then, the structural similarity measure has been applied on the candidate resulting states {e_1, e_2 ... e_n} in the second level. As a result, our method will output the state which has the highest confidence. Moreover, our resulting mapping state {e_f} is measured based on its rich structure on one hand and the greatest number of corresponding concepts between the source ontologies on the other hand.

2 Formulation of the Framework

In this section, we provide the definitions for the main components of our framework. These definitions give the meaning of our notations such as, what are the entities we are referring to, the relationship matrix that gives the basis to compute the similarity matrix, the matching matrix, the matching space, and in the subsection we introduce the structure-based similarity measure.

We describe the mapping problem as identifying pairs of similar nodes (also called vertices) in the input ontologies modeled as labeled directed graphs. The nodes in an input graph correspond to entities in ontologies, and the edges indicate the relationships between the pair of nodes they connect. The labels indicate the kind of relationship, e.g. "domain" or "range." In this study, we limit ourselves to finding mappings for classes and relationships only.

Definition 1 (Entity-relationships). Let S be a source ontology, T be a target ontology. We use E^S = {s_1, s_2,..., s_n} and E^T = {t_1, t_2,..., t_m} to denote the set of entities in S and T, respectively. Entity refers to classes, properties, or individuals for which we want to find matches in the input ontologies. We use $R(r_{ij})$, defined below, to denote the relationship between entities s_i and t_j. We use r_{ij} to denote a matching degree between s_i and t_j.

Definition 2 (Relationship Matrix). This relational matrix, denoted as $R(r_{ij})$, represents the relationship between ontologies S and T, i.e., r_{ij} includes indicates the similarity between concept s_i in S and concept t_j in T. Using R, we define another relational matrix, called the *similarity matrix*, which captures a different relationship between S and T, defined as follows.

Definition 3 (Similarity Matrix). This relational matrix, denoted $L(l_{ij})$, includes entries in [0,1], called the *similarity coefficients*, representing the degree of similarity between s_i and t_j. Both R and L are n×m matrices.

Definition 4 (Matching Matrix). A matching matrix, denoted $Map_{0\text{-}1}$, is a 0-1 matrix with dimension n×m and with entries $r_{ij} \in \{0,1\}$. If $r_{ij} = 1$, it means that S_i and t_j are "matchable." They are unmatchable if $r_{ij} = 0$.

Definition 5 (Matching Space). All the possible assignments for the matching matrix form a *matching space*, also called the *mapping space*. Every assignment is a state in the matching space. The state represents a solution of ontology matching. The following example illustrates the above concepts and terms.

Example 1. Let S and T be the input ontologies, and $E^S=\{s_1,s_2,\ldots,s_n\}$ and $E^T=\{t_1, t_2,\ldots,t_m\}$be the sets of entitie. A matching matrix $Map_{0\text{-}1}$ indicates the similarity relation between the elements of E^S and E^T. The number of relationship matrices $Map_{0\text{-}1}$ is $2^{n\text{x}m}$, i.e., the matching space has $2^{n\text{x}m}$ states. These matrices form the matching space. For instance, when $Map_{0\text{-}1}$ is 2×2, the matching space would have 16 states. Some of these mapping states are as follows, in which the rows are entities in S and the columns are entities in T. E.g., the first matrix indicates no mapping. The third matrix below, it indicates that entity s_1 is matched with t_1 or t_2, and s_2 is matched with t_2, etc.

$$\left(\begin{bmatrix}0&0\\0&0\end{bmatrix},\begin{bmatrix}1&0\\0&0\end{bmatrix},\ldots,\begin{bmatrix}1&1\\0&1\end{bmatrix},\begin{bmatrix}1&1\\1&1\end{bmatrix}\right).$$

2.1 Tradeoff Between Structure and Size of the Mapping States

Many similarity measures have been introduced for a set of keywords representing a text. For example, the Dice coefficient, the Jaccard coefficient, the Cosine coefficient [21], etc. The Dice coefficient is defined as follows:

$$S_{T_1,T_2} = (2|T_1 \cap T_2|)/(|T_1| + |T_2|)\ .$$

where $|T_i|$ is the number of terms in set T_i, and $|T_1 \cap T_2|$ is the number of common terms in T_1 and T_2. We will use this as the similarity measure in our work.

Let O_1 and O_2 be a pair of ontologies represented as labeled graphs, and O_{MMA} be the ontology induced by the similarity result S_{MMA} obtained by applying the basic MMA match algorithm (which combines the similarity measures in a single step/level operation). Let S_{strc} be the structural similarity measure S, calculated as follows, which defines the similarities between the concepts provided by O_{MMA} and those in the original ontologies O_1 and O_2.

$$S_{strc} = 2\,|r(O_{MMA})| / (|r(O_{MMA}(O_1))| + |r(O_{MMA}(O_2))|)\ .$$

where $|r(O_{MMA})|$ is the number of relationships in ontology O_{MMA}, and $|r(O_{MMA}(O_i))|$ is the number of relationships in the immediate neighborhood of O_{MMA} in O_i. This neighborhood of O_{MMA} consists of the relationships of O_i with at least one end (one of the edge's end) belonging to O_{MMA}.

We view S_{strc} as a complementary measure to the output of MMA, applied in the second level. This is justified as follows.

- The structure similarity S_{strc} is mainly based on the presence of common concepts between the matched ontologies induced by the states calculated by MMA, and
- the similarity degree between the matched ontologies may still exist, even when there is no structural match in the result of MMA, i.e., when $S_{strc} = 0$.

Accordingly, the combined similarity measure S is relative to S_{MMA}, and should not be zero in case $S_{strc} = 0$. We further "smooth" the effect of S_{strc} as follws:

$$S = S_{MMA} + (x * S_{strc}) , \quad where \quad x = (1 - S_{MMA}) .$$

In the combined similarity S, suppose $S_{strc}= 0$. This then means S just depends on the similarity measure of MMA. On the other hand, if $S_{strc} = 1$, the neighborhood of the concepts matched by MMA is the same, and consequently S will take the maximum value, and since $1= S_{MMA} + x$, we have that $x = 1 - S_{MMA}$, representing the complementary part of information described in the relationships among the concepts in a desired state found by MMA.

As we do not want to miss a matching state found which includes a large number of concepts matched, S_{MMA} provides possible good matches in the input ontologies together with the similarity degrees. The extended method will determine the same collection of matched states, but with better differentiation among them by taking into account the structural measures in the second level. An extension of this two level method to a multi-level method is straightforward, when the user can identify which measures could or should be applied at which level.

3 Structure-Based Multi Level Matching Algorithm

Now we study various matching spaces, and show how to construct the matching spaces. Then, we describe an algorithm to solve OMP, using MLMA.

3.1 The MLMA Algorithm

There are many algorithms for matching spaces. The notion of multispace "combines" all desired spaces into a single unified space. By searching from space to space, the matching algorithm can find a reasonable solution eventually. The main idea of the proposed Multi-Level algorithm is shown in Fig. 5.

The algorithm is mainly divided into three phases. In phase 1, which is the initialization phase, an initial assignment for the matching matrix *Map* is provided, as well as the functions of similarity to evaluate the relationship matrix. In phase 2 of MMA, which is the search phase, it is an iterative refinement for the *Map* matrices. In phase 3, the resulting mapping states from MMA will be qualified based on the connectivity among their concepts. Then, the best possible final state will be offered to the user.

The algorithm iteratively constructs matching spaces for entities of both S and T (see illustrative example in the next section). Then, the *Map* matrices will be evaluated according to the re/used spaces such as name and linguistic spaces, and finally the mapping state with the highest evaluation value will be offered to the user. If we only search one matching space, the algorithm behaves and computes as a single matcher; otherwise, it is indeed a multi-matcher. This design is useful as it provides a flexible and convenient way to use various relevant information about input ontologies, and to combine feasible mapping methods to obtain a far better matching result than the results obtained by each individual method. The method can employ any desired search algorithm.

```
Given: Two ontologies S and T
Output: The mapping result between S and T
Phase 1 Initialization
     Design an initial assignment matching matrix.
     /* For example, let Map be the zero matrix,
     or let diagonal elements in Map be equal to 1, and so
     on.*/
     Use the similarity functions to evaluate similarity or
     relationship matrix.
Phase 2 Search Matching Space
    begin
      Enter an active search space
      /* such as the name matching space */
      Evaluate an intermediate matching state
      /* more better matching results */
       begin
            Enter another active search space
            /* such as the linguistic matching space */
            Evaluate a better intermediate matching state
              Begin
                 ...
                /* various available matching spaces,
              i.e. many feasible matching methods */
               end;
         end;
         if the intermediate matching state is not
            the final solution
            /* the matching result does not satisfy
               the evaluation function */
              then use it as an initial solution in the
             next iteration;
               if the matching instance satisfies the
                  evaluation function
                 then return the final solution
     end;

Phase 3 Apply the Complementary measures
            /* Apply the structure similarity measure
               to the output of phase 2. */
```

Fig. 5. The Multi-Level Match Algorithm

3.2 Multiple Matching Spaces

Matching spaces are distinguished by diverse similarity measures. Moreover, the different kinds of similarity measures between the entities of the ontologies use different methods to compare the similarity of two ontologies. Accordingly, we construct the similarity matrices and matching spaces. Furthermore, different relation spaces are built on the result of using different methods of measuring similarity. These methods can be classified as follows (see [12] for more detailed explanation).

- *String similarity.* These methods are based on the hypothesis that concepts and property names representing semantic similarity will have similar syntactic features. The Levenshtein distance is the simplest implementation of string distance.
- *Linguistic similarity.* This is an extension of string similarity measures with some semantics. For example, considering the synonyms based on some specific thesauri, e.g., WordNet.

此为试读,需要完整PDF请访问: www.ertongbook.com