Journal Subline

LNCS 3360

Journal on
Data Semantics II

Stefano Spaccapietra
Editor-in-Chief

Springer

Stefano Spaccapietra
Elisa Bertino Sushil Jajodia
Roger King Dennis McLeod
Maria E. Orlowska Leon Strous (Eds.)

Journal on Data Semantics II

Springer

Volume Editors

Stefano Spaccapietra
E-mail: stefano.spaccapietra@epfl.ch

Elisa Bertino
E-mail: bertino@cs.purdue.edu

Sushil Jajodia
E-mail: jajodia@gmu.edu

Roger King
E-mail: roger@cs.colorado.edu

Dennis McLeod
E-mail: mcleod@usc.edu

Maria E. Orlowska
E-mail: maria@itee.uq.edu.au

Leon Strous
E-mail: strous@iae.nl

Library of Congress Control Number: 2004117338

CR Subject Classification (1998): H.2, H.3, I.2, H.4, C.2

ISSN 0302-9743
ISBN 3-540-24208-2 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Boller Mediendesign
Printed on acid-free paper   SPIN: 11371861   06/3142   5 4 3 2 1 0

Lecture Notes in Computer Science 3360

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

The LNCS Journal on Data Semantics

Computerized information handling has changed its focus from centralized data management systems to decentralized data exchange facilities. Modern distribution channels, such as high-speed Internet networks and wireless communication infrastructures, provide reliable technical support for data distribution and data access, materializing the new popular idea that data may be available to anybody, anywhere, anytime. However, providing huge amounts of data on request often turns into a counterproductive service, making the data useless because of poor relevance or inappropriate levels of detail. Semantic knowledge is the essential missing piece that allows the delivery of information that matches user requirements. Semantic agreement, in particular, is essential to meaningful data exchange.

Semantic issues have long been open issues in data and knowledge management. However, the boom in semantically poor technologies, such as the Web and XML, has prompted a renewed interest in semantics. Conferences on the Semantic Web, for instance, attract crowds of participants, while ontologies on its own has become a hot and popular topic in the database and artificial intelligence communities.

Springer's LNCS Journal on Data Semantics aims at providing a highly visible dissemination channel for the most remarkable work that in one way or another addresses research and development on issues related to the semantics of data. The target domain ranges from theories supporting the formal definition of semantic content to innovative domain-specific applications of semantic knowledge. This publication channel should be of highest interest to researchers and advanced practitioners working on the Semantic Web, interoperability, mobile information services, data warehousing, knowledge representation and reasoning, conceptual database modeling, ontologies, and artificial intelligence.

Topics of relevance to this journal include:

- semantic interoperability, semantic mediators
- ontologies
- ontology, schema and data integration, reconciliation and alignment
- multiple representations, alternative representations
- knowledge representation and reasoning
- conceptualization and representation
- multimodel and multiparadigm approaches
- mappings, transformations, reverse engineering
- metadata
- conceptual data modeling
- integrity description and handling
- evolution and change
- Web semantics and semistructured data
- semantic caching

- data warehousing and semantic data mining
- spatial, temporal, multimedia and multimodal semantics
- semantics in data visualization
- semantic services for mobile users
- supporting tools
- applications of semantic-driven approaches

These topics are to be understood as specifically related to semantic issues. Contributions submitted to the journal and dealing with semantics of data will be considered even if they are not within the topics in the list.

While the physical appearance of the journal issues looks like the books from the well-known Springer LNCS series, the mode of operation is that of a journal. Contributions can be freely submitted by authors and are reviewed by the Editorial Board. Contributions may also be invited, and nevertheless carefully reviewed, as in the case of issues that contain extended versions of best papers from major conferences addressing data semantics issues. Special issues, focusing on a specific topic, are coordinated by guest editors once the proposal for a special issue is accepted by the Editorial Board. Finally, it is also possible that a journal issue be devoted to a single text.

The journal published its first volume in 2003. This is the second volume, and it will be followed by three volumes to appear in 2005.

The Editorial Board comprises one Editor-in-Chief (with overall responsibility) and several members. The editor-in-chief has a four-year mandate to run the journal. Members of the board have three-year mandates. Mandates are renewable. More members may be added to the Editorial Board as appropriate

We are happy to welcome you into our readership and authorship, and hope we will share this privileged contact for a long time.

Stefano Spaccapietra
Editor-in-Chief

JoDS Volume 2 – Guest Editorial

Conferences provide researchers with the fastest way to disseminate their ideas and results to a selected community of other researchers in the same domain. Conferences, however, must enforce limitations in the sizes of the written contributions as well as in the time allocated for the on-site presentations of the contributions. They also have limited audiences, although some publishers such as Springer have a dissemination scheme that brings conference proceedings to much wider audiences than just the actual participants at the conferences.

Publication of an extended version of a conference paper is a much appreciated opportunity for researchers to widely disseminate a significantly improved presentation of their work, where they can develop the appropriate motivations, reasoning, results and comparative analysis. To foster dissemination of the best ideas and results, the Journal on Data Semantics (JoDS) pursues a policy that includes annually publishing extended versions of the best papers from selected conferences whose scope encompasses or intersects the scope of the journal.

The selection for this issue comprises the International Conference on Ontologies, Databases and Applications of Semantics (ODBASE), the International Conference on Cooperative Information Systems (COOPIS), and the IFIP TC11 WG11.5 Working Conference on Integrity and Internal Control in Information Systems (IICIS). Papers from these conferences were selected based on their quality, relevance, and significance, and the viability of extending their results. All extended papers were subject to a stringent review process and the authors were required to respond to all concerns expressed by the reviewers before papers were accepted.

Four papers, showing consistently high reviews from the program committee, were selected among those presented at the *Ontologies, Databases and Applications of Semantics (ODBase)* conference, held in Catania, Italy, November 4–6, 2003. Three of the papers have to do with the construction and maintenance of ontologies and structured taxonomies. *Incrementally Maintaining Materializations of Ontologies Stored in Logic Databases* (by Raphael Volz, Steffen Staab, and Boris Motik) presents a method for propagating changes made to an ontology; the technique is broadly applicable, as it is compatible with any ontology language that can be translated into Datalog programs. *Ontology Translation on the Semantic Web* (by Dejing Dou, Drew McDermott, and Peishen Qi) addresses the highly important problem of resolving terminology differences between related ontologies; the technique manages syntactic as well as semantic translations. *Compound Term Composition Algebra: the Semantics* (by Yannis Tzitzikas, Anastasia Analyti, and Nicolas Spyratos) presents an elegant, formal algebra for specifying the valid compound terms in a taxonomy.

The fourth paper, *Dynamic Pattern Mining: an Incremental Data Clustering Approach* (by Seokkyung Chung and Dennis McLeod) addresses one of the central problems facing users of data mines – the incremental maintenance of a data mine that is constantly updated. The paper deals specifically with services that provide integrated access to news articles; the method described in the paper is simple yet semantically powerful and quite efficient.

The volume continues with two papers that are comprehensive descriptions of the topics of the two top-rated papers that appeared in the *CoopIS* portion of the proceedings of the conference triad *On the Move to Meaningful Internet Systems*, November 2003. The first paper, *A Knowledge Network Approach for Implementing Active Virtual Marketplaces*, by Minsoo Lee, Stanley Su, and Herman Lam, presents a network approach for implementing virtual marketplaces: bringing buyers and sellers cooperatively together. The paper focuses on an infrastructure that enables sharing of knowledge over the Web and thus effectively supports the formation of virtual marketplaces on the Web. The concept of an active virtual marketplace is realized using this infrastructure by allowing buyers and sellers to specify their knowledge in the form of events, triggers, and rules. The knowledge network can actively distribute and process these knowledge elements to help buyers and sellers to locate and interact with each other.

The second paper, *Stream Integration Techniques for Grid Monitoring*, by Andy Cooke, Alasdair Gray, and Werner Nutt, focuses on a technique for providing information about the status of a cooperative computation grid by utilizing database integration techniques. This novel approach provides an infrastructure for publishing and querying grid monitoring data. Emphasis is placed on the use of the technique for distributed sets of data streams, which provide information about the changes over time of a data source. The concepts and mechanisms devised can also be applied more generally where there is a need for publishing and querying information in a distributed manner.

Finally, the volume contains two papers originally presented at the *6th IFIP TC 11 WG 11.5 Working Conference on Integrity and Internal Control in Information Systems*, which was held November 13–14, 2003 in Lausanne, Switzerland. Traditionally, access controls have been used to limit the availability of data to users; however, they do not protect unauthorized disclosure of sensitive information from careless or malicious insiders with authorized access to the system. The first paper *Information Release Control: a Learning-Based Architecture*, by Claudio Bettini, X. Sean Wang, and Sushil Jajodia, explores the information release control paradigm, which is based on checking data when they are being released across organizational boundaries. Rather than relying simply on source/destination addresses, as in current firewall systems, or on simple "dirty word" matching as in current filtering software, the checking process analyzes the semantics of the released data. This paper formalizes this process and presents the architecture of a system that incorporates a module for learning release constraints.

Nowadays, surveillance devices such as video cameras and microphones have become commonplace in our society. The second paper, *Enforcing Semantics-Aware Security in Multimedia Surveillance*, by Naren Kodali, Csilla Farkas, and Duminda Wijesekera, considers the surveillance data flowing at secured facilities such as airports, nuclear power plants, and national laboratories. Typically, different parts of such facilities have different degrees of sensitivity. Likewise, human guards are categorized according to their rights of access to various locations within the facilities. The main security requirement is that the guards can view data gathered from locations whose sensitivity is consistent with their access rights. This paper shows how to model the surveillance requirements using the synchronized multimedia integration language (SMIL) with appropriate multilevel security enhancements.

Guest Editors

ODBASE	Roger (Buzz) King, University of Colorado at Boulder, USA Maria Orlowska, University of Queensland, Australia
COOPIS	Elisa Bertino, Purdue University, USA Dennis McLeod, University of Southern California, Los Angeles, USA
IICIS	Sushil Jajodia, George Mason University, USA Leon Strous, De Nederlandsche Bank, The Netherlands

We are pleased to express our gratitude to the reviewers, who invested much of their time in careful analysis and evaluation of the submissions:

Dave Abel, CSIRO, Australia
Karl Aberer, École Polytechnique Fédérale de Lausanne, Switzerland
John Carlis, University of Minnesota, USA
Tiziana Catarci, University of Rome, Italy
Brian Cooper, Georgia Institute of Technology, USA
Guido Governatori, University of Queensland, Australia
Michael Kifer, Stonybrook State University of New York, USA
Dik Lee, University of Science and Technology, Hong Kong, China
Qing Li, City University of Hong Kong, Hong Kong, China
Leo Mark, Georgia Institute of Technology, USA
Ravi Mukkamala, Old Dominion University, USA
Erich Neuhold, Darmstadt University of Technology, Germany
Brajendra Panda, University of Arkansas, Fayetteville, USA
Evagelia Pitoura, University of Ioannina, Greece
Amit Sheth, University of Georgia, USA
Antonio Si, Oracle, USA
Steffen Staab, University of Koblenz-Landau, Germany
Sean Wang, University of Vermont, USA
Chao Yao, George Mason University, USA
Roger Zimmermann, University of Southern California, USA

JoDS Editorial Board[1]

[1] The late Yahiko Kambayashi (Kyoto University, Japan) was a member of the JoDS Editorial Board.

Lecture Notes in Computer Science

For information about Vols. 1–3259

please contact your bookseller or Springer

Vol. 3385: R. Cousot (Ed.), Verification, Model Checking, and Abstract Interpretation. XII, 483 pages. 2004.

Vol. 3382: J. Odell, P. Giorgini, J.P. Müller (Eds.), Agent-Oriented Software Engineering V. X, 239 pages. 2004.

Vol. 3381: M. Bieliková, B. Charon-Bost, O. Sýkora, P. Vojtáš (Eds.), SOFSEM 2005: Theory and Practice of Computer Science. XV, 428 pages. 2004.

Vol. 3363: T. Eiter, L. Libkin (Eds.), Database Theory - ICDT 2005. XI, 413 pages. 2004.

Vol. 3360: S. Spaccapietra, E. Bertino, S. Jajodia, R. King, D. McLeod, M.E. Orlowska, L. Strous (Eds.), Journal on Data Semantics II. XI, 223 pages. 2004.

Vol. 3358: J. Cao, L.T. Yang, M. Guo, F. Lau (Eds.), Parallel and Distributed Processing and Applications. XXIV, 1058 pages. 2004.

Vol. 3356: G. Das, V.P. Gulati (Eds.), Intelligent Information Technology. XII, 428 pages. 2004.

Vol. 3353: J. Hromkovič, M. Nagl, B. Westfechtel (Eds.), Graph-Theoretic Concepts in Computer Science. XI, 404 pages. 2004.

Vol. 3350: M. Hermenegildo, D. Cabeza (Eds.), Practical Aspects of Declarative Languages. VIII, 269 pages. 2004.

Vol. 3348: A. Canteaut, K. Viswanathan (Eds.), Progress in Cryptology - INDOCRYPT 2004. XIV, 431 pages. 2004.

Vol. 3347: R.K. Ghosh, H. Mohanty (Eds.), Distributed Computing and Internet Technology. XX, 472 pages. 2004.

Vol. 3344: J. Malenfant, B.M. Østvold (Eds.), Object-Oriented Technology. ECOOP 2004 Workshop Reader. VIII, 215 pages. 2004.

Vol. 3342: E. Şahin, W.M. Spears (Eds.), Swarm Robotics. X, 175 pages. 2004.

Vol. 3341: R. Fleischer, G. Trippen (Eds.), Algorithms and Computation. XVII, 935 pages. 2004.

Vol. 3340: C.S. Calude, E. Calude, M.J. Dinneen (Eds.), Developments in Language Theory. XI, 431 pages. 2004.

Vol. 3339: G.I. Webb, X. Yu (Eds.), AI 2004: Advances in Artificial Intelligence. XXII, 1272 pages. 2004. (Subseries LNAI).

Vol. 3338: S.Z. Li, J. Lai, T. Tan, G. Feng, Y. Wang (Eds.), Advances in Biometric Person Authentication. XVIII, 699 pages. 2004.

Vol. 3337: J.M. Barreiro, F. Martin-Sanchez, V. Maojo, F. Sanz (Eds.), Biological and Medical Data Analysis. XI, 508 pages. 2004.

Vol. 3336: D. Karagiannis, U. Reimer (Eds.), Practical Aspects of Knowledge Management. X, 523 pages. 2004. (Subseries LNAI).

Vol. 3334: Z. Chen, H. Chen, Q. Miao, Y. Fu, E. Fox, E.-p. Lim (Eds.), Digital Libraries: International Collaboration and Cross-Fertilization. XX, 690 pages. 2004.

Vol. 3333: K. Aizawa, Y. Nakamura, S. Satoh (Eds.), Advances in Multimedia Information Processing - PCM 2004, Part III. XXXV, 785 pages. 2004.

Vol. 3332: K. Aizawa, Y. Nakamura, S. Satoh (Eds.), Advances in Multimedia Information Processing - PCM 2004, Part II. XXXVI, 1051 pages. 2004.

Vol. 3331: K. Aizawa, Y. Nakamura, S. Satoh (Eds.), Advances in Multimedia Information Processing - PCM 2004, Part I. XXXVI, 667 pages. 2004.

Vol. 3329: P.J. Lee (Ed.), Advances in Cryptology - ASIACRYPT 2004. XVI, 546 pages. 2004.

Vol. 3328: K. Lodaya, M. Mahajan (Eds.), FSTTCS 2004: Foundations of Software Technology and Theoretical Computer Science. XVI, 532 pages. 2004.

Vol. 3326: A. Sen, N. Das, S.K. Das, B.P. Sinha (Eds.), Distributed Computing - IWDC 2004. XIX, 546 pages. 2004.

Vol. 3323: G. Antoniou, H. Boley (Eds.), Rules and Rule Markup Languages for the Semantic Web. X, 215 pages. 2004.

Vol. 3322: R. Klette, J. Žunić (Eds.), Combinatorial Image Analysis. XII, 760 pages. 2004.

Vol. 3321: M.J. Maher (Ed.), Advances in Computer Science - ASIAN 2004. XII, 510 pages. 2004.

Vol. 3320: K.-M. Liew, H. Shen, S. See, W. Cai (Eds.), Parallel and Distributed Computing: Applications and Technologies. XXIV, 891 pages. 2004.

Vol. 3316: N.R. Pal, N.K. Kasabov, R.K. Mudi, S. Pal, S.K. Parui (Eds.), Neural Information Processing. XXX, 1368 pages. 2004.

Vol. 3315: C. Lemaître, C.A. Reyes, J.A. González (Eds.), Advances in Artificial Intelligence – IBERAMIA 2004. XX, 987 pages. 2004. (Subseries LNAI).

Vol. 3314: J. Zhang, J.-H. He, Y. Fu (Eds.), Computational and Information Science. XXIV, 1259 pages. 2004.

Vol. 3312: A.J. Hu, A.K. Martin (Eds.), Formal Methods in Computer-Aided Design. XI, 445 pages. 2004.

Vol. 3311: V. Roca, F. Rousseau (Eds.), Interactive Multimedia and Next Generation Networks. XIII, 287 pages. 2004.

Vol. 3309: C.-H. Chi, K.-Y. Lam (Eds.), Content Computing. XII, 510 pages. 2004.

Vol. 3308: J. Davies, W. Schulte, M. Barnett (Eds.), Formal Methods and Software Engineering. XIII, 500 pages. 2004.

Vol. 3307: C. Bussler, S.-k. Hong, W. Jun, R. Kaschek, D.. Kinshuk, S. Krishnaswamy, S.W. Loke, D. Oberle, D. Richards, A. Sharma, Y. Sure, B. Thalheim (Eds.), Web Information Systems – WISE 2004 Workshops. XV, 277 pages. 2004.

Vol. 3306: X. Zhou, S. Su, M.P. Papazoglou, M.E. Orlowska, K.G. Jeffery (Eds.), Web Information Systems – WISE 2004. XVII, 745 pages. 2004.

Vol. 3305: P.M.A. Sloot, B. Chopard, A.G. Hoekstra (Eds.), Cellular Automata. XV, 883 pages. 2004.

Vol. 3303: J.A. López, E. Benfenati, W. Dubitzky (Eds.), Knowledge Exploration in Life Science Informatics. X, 249 pages. 2004. (Subseries LNAI).

Vol. 3302: W.-N. Chin (Ed.), Programming Languages and Systems. XIII, 453 pages. 2004.

Vol. 3300: L. Bertossi, A. Hunter, T. Schaub (Eds.), Inconsistency Tolerance. VII, 295 pages. 2004.

Vol. 3299: F. Wang (Ed.), Automated Technology for Verification and Analysis. XII, 506 pages. 2004.

Vol. 3298: S.A. McIlraith, D. Plexousakis, F. van Harmelen (Eds.), The Semantic Web – ISWC 2004. XXI, 841 pages. 2004.

Vol. 3296: L. Bougé, V.K. Prasanna (Eds.), High Performance Computing - HiPC 2004. XXV, 530 pages. 2004.

Vol. 3295: P. Markopoulos, B. Eggen, E. Aarts, J.L. Crowley (Eds.), Ambient Intelligence. XIII, 388 pages. 2004.

Vol. 3294: C.N. Dean, R.T. Boute (Eds.), Teaching Formal Methods. X, 249 pages. 2004.

Vol. 3293: C.-H. Chi, M. van Steen, C. Wills (Eds.), Web Content Caching and Distribution. IX, 283 pages. 2004.

Vol. 3292: R. Meersman, Z. Tari, A. Corsaro (Eds.), On the Move to Meaningful Internet Systems 2004: OTM 2004 Workshops. XXIII, 885 pages. 2004.

Vol. 3291: R. Meersman, Z. Tari (Eds.), On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE, Part II. XXV, 824 pages. 2004.

Vol. 3290: R. Meersman, Z. Tari (Eds.), On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE, Part I. XXV, 823 pages. 2004.

Vol. 3289: S. Wang, K. Tanaka, S. Zhou, T.W. Ling, J. Guan, D. Yang, F. Grandi, E. Mangina, I.-Y. Song, H.C. Mayr (Eds.), Conceptual Modeling for Advanced Application Domains. XXII, 692 pages. 2004.

Vol. 3288: P. Atzeni, W. Chu, H. Lu, S. Zhou, T.W. Ling (Eds.), Conceptual Modeling – ER 2004. XXI, 869 pages. 2004.

Vol. 3287: A. Sanfeliu, J.F. Martínez Trinidad, J.A. Carrasco Ochoa (Eds.), Progress in Pattern Recognition, Image Analysis and Applications. XVII, 703 pages. 2004.

Vol. 3286: G. Karsai, E. Visser (Eds.), Generative Programming and Component Engineering. XIII, 491 pages. 2004.

Vol. 3285: S. Manandhar, J. Austin, U.B. Desai, Y. Oyanagi, A. Talukder (Eds.), Applied Computing. XII, 334 pages. 2004.

Vol. 3284: A. Karmouch, L. Korba, E.R.M. Madeira (Eds.), Mobility Aware Technologies and Applications. XII, 382 pages. 2004.

Vol. 3283: F.A. Aagesen, C. Anutariya, V. Wuwongse (Eds.), Intelligence in Communication Systems. XIII, 327 pages. 2004.

Vol. 3282: V. Guruswami, List Decoding of Error-Correcting Codes. XIX, 350 pages. 2004.

Vol. 3281: T. Dingsøyr (Ed.), Software Process Improvement. X, 207 pages. 2004.

Vol. 3280: C. Aykanat, T. Dayar, İ. Körpeoğlu (Eds.), Computer and Information Sciences - ISCIS 2004. XVIII, 1009 pages. 2004.

Vol. 3279: G.M. Voelker, S. Shenker (Eds.), Peer-to-Peer Systems III. XI, 300 pages. 2004.

Vol. 3278: A. Sahai, F. Wu (Eds.), Utility Computing. XI, 272 pages. 2004.

Vol. 3275: P. Perner (Ed.), Advances in Data Mining. VIII, 173 pages. 2004. (Subseries LNAI).

Vol. 3274: R. Guerraoui (Ed.), Distributed Computing. XIII, 465 pages. 2004.

Vol. 3273: T. Baar, A. Strohmeier, A. Moreira, S.J. Mellor (Eds.), <<UML>> 2004 - The Unified Modelling Language. XIII, 454 pages. 2004.

Vol. 3272: L. Baresi, S. Dustdar, H. Gall, M. Matera (Eds.), Ubiquitous Mobile Information and Collaboration Systems. VIII, 197 pages. 2004.

Vol. 3271: J. Vicente, D. Hutchison (Eds.), Management of Multimedia Networks and Services. XIII, 335 pages. 2004.

Vol. 3270: M. Jeckle, R. Kowalczyk, P. Braun (Eds.), Grid Services Engineering and Management. X, 165 pages. 2004.

Vol. 3269: J. Lopez, S. Qing, E. Okamoto (Eds.), Information and Communications Security. XI, 564 pages. 2004.

Vol. 3268: W. Lindner, M. Mesiti, C. Türker, Y. Tzitzikas, A. Vakali (Eds.), Current Trends in Database Technology - EDBT 2004 Workshops. XVIII, 608 pages. 2004.

Vol. 3267: C. Priami, P. Quaglia (Eds.), Global Computing. VIII, 377 pages. 2004.

Vol. 3266: J. Solé-Pareta, M. Smirnov, P.V. Mieghem, J. Domingo-Pascual, E. Monteiro, P. Reichl, B. Stiller, R.J. Gibbens (Eds.), Quality of Service in the Emerging Networking Panorama. XVI, 390 pages. 2004.

Vol. 3265: R.E. Frederking, K.B. Taylor (Eds.), Machine Translation: From Real Users to Research. XI, 392 pages. 2004. (Subseries LNAI).

Vol. 3264: G. Paliouras, Y. Sakakibara (Eds.), Grammatical Inference: Algorithms and Applications. XI, 291 pages. 2004. (Subseries LNAI).

Vol. 3263: M. Weske, P. Liggesmeyer (Eds.), Object-Oriented and Internet-Based Technologies. XII, 239 pages. 2004.

Vol. 3262: M.M. Freire, P. Chemouil, P. Lorenz, A. Gravey (Eds.), Universal Multiservice Networks. XIII, 556 pages. 2004.

Vol. 3261: T. Yakhno (Ed.), Advances in Information Systems. XIV, 617 pages. 2004.

Vol. 3260: I.G.M.M. Niemegeers, S.H. de Groot (Eds.), Personal Wireless Communications. XIV, 478 pages. 2004.

Table of Contents

Incrementally Maintaining Materializations of Ontologies Stored in Logic Databases

Raphael Volz[1,2], Steffen Staab[1], and Boris Motik[2]

[1] Institute AIFB, University of Karlsruhe, Germany
lastname@aifb.uni-karlsruhe.de
[2] FZI, University of Karlsruhe, Germany
volz@fzi.de

Abstract. This article presents a technique to incrementally maintain materializations of ontological entailments. Materialization consists in precomputing and storing a set of implicit entailments, such that frequent and/or crucial queries to the ontology can be solved more efficiently. The central problem that arises with materialization is its maintenance when axioms change, viz. the process of propagating changes in explicit axioms to the stored implicit entailments.

When considering rule-enabled ontology languages that are operationalized in logic databases, we can distinguish two types of changes. Changes to the ontology will typically manifest themselves in changes to the rules of the logic program, whereas changes to facts will typically lead to changes in the extensions of logical predicates. The incremental maintenance of the latter type of changes has been studied extensively in the deductive database context and we apply the technique proposed in [30] for our purpose. The former type of changes has, however, not been tackled before.

In this article we elaborate on our previous papers [32, 33], which extend the approach of [30] to deal with changes in the logic program. Our approach is not limited to a particular ontology language but can be generally applied to arbitrary ontology languages that can be translated to Datalog programs, i.e. such as O-Telos, F-Logic [16] RDF(S), or Description Logic Programs [34].

1 Introduction

Germane to the idea of the Semantic Web are the capabilities to assert facts and to derive new facts from the asserted facts using the semantics specified by an ontology. Both current building blocks of the Semantic Web, RDF [13] and OWL [21], define how to assert facts and specify how new facts should be derived from stated facts.

The necessary derivation of entailed information from asserted information is usually achieved at the time clients issue queries to inference engines such as logic databases. Situations where queries are frequent or the procedure to derive entailed information is time consuming and complex typically lead to low performance. Materialization can be used to increase the performance at

S. Spaccapietra et al. (Eds.): Journal on Data Semantics II, LNCS 3360, pp. 1–34, 2004.

query time by making entailed information explicit upfront. Thereby, the recomputation of entailed information for every single query is avoided.

Materialization has been applied successfully in many applications where reading access to data is predominant. For example, data warehouses usually apply materialization techniques to make *online* analytical processing possible. Similarly, most Web portals maintain cached web pages to offer fast access to dynamically generated web pages.

We conjecture that reading access to ontologies is predominant in the Semantic Web and other ontology-based applications, hence materialization seems to be a promising technique for fast processing of queries on ontologies.

Materialization is particularly promising for the currently predominant approach of aggregating distributed information into a central knowledge base (cf. [8, 14, 31, 20]). For example, the OntoWeb[3] Semantic portal [28] employs a *syndicator* (cf. Figure 1), which regularly visits sites specified by community members and transfers the detected updates into a central knowledge base in a batch process. Hence, the knowledge base remains unchanged between updates for longer periods of time.

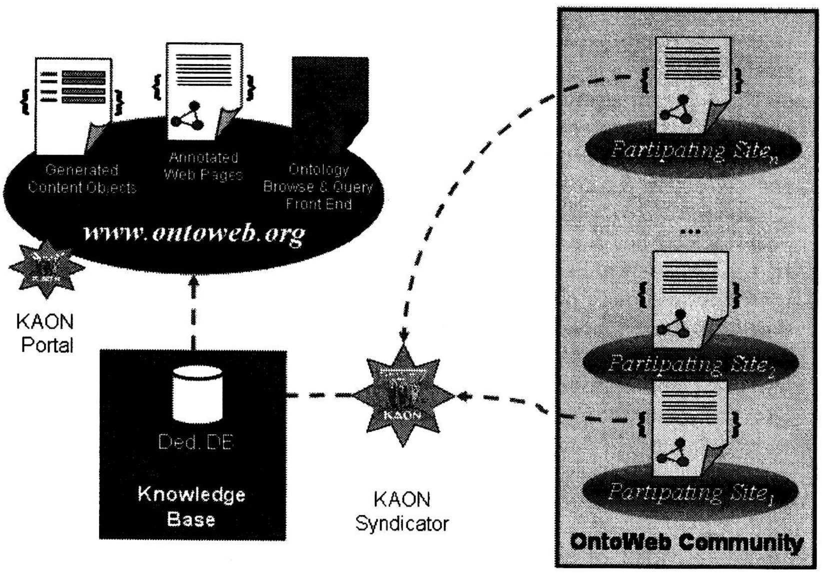

Fig. 1. OntoWeb Architecture

The OntoWeb portal, however, provides answers to queries issued on the knowledge base whenever visitors browse the portal content. This is due to the fact that most queries are hard-coded into the definition of dynamic Web pages,

[3] http://www.ontoweb.org/.

which are generated for every request. In applications such as OntoWeb, materialization turns out to be a sine qua non.[4]

Central to materialization approaches is the issue of maintaining a materialization when changes occur. This issue can be handled by simply recomputing the whole materialization. However, as the computation of the materialization is often complex and time consuming, it is desirable to apply more efficient techniques in practice, i.e. to *incrementally* maintain a materialization.

1.1 Contribution

We present a technique for the incremental maintenance of materialized ontologies. Our technique can be applied to a wide range of ontology languages, namely those that can be axiomatized by a set of rules[5].

The challenge that has not been tackled before comes from the fact that updates of ontology definitions are equivalent to the updates and new definitions of rules, whereas existing maintenance techniques only address the update of ground facts.

To cope with changing rules, our solution extends a declarative algorithm for the incremental maintenance of views [30] that was developed in the deductive database context. We show the feasibility of our solution in a performance evaluation.

1.2 Organization

The remainder of the article is organized as follows: Section 2 reviews how current Web ontology languages such as RDF(S) and OWL interplay with rules. Section 3 presents the underlying principles which are applied to achieve incremental maintenance of a materialization. Section 4 recapitulates the incremental maintenance algorithm presented in [30] , presents a novel modular rewriting algorithm based on generator functions and shows how this algorithm deals with changes to facts. Section 5 extends this algorithm to deal with changing rules as they result from changes in the ontology. Section 6 sketches how the developed techniques can be applied in implementations of RDF rule languages. Section 7 describes our prototypical implementation. Section 8 performs a performance analysis and shows the benefits of our approach. Section 10 summarizes our contribution and discusses further uses.

2 Web Ontology Languages and Logic Databases

In the brief history of the Semantic Web, most applications, e.g. [7], have implemented the logical entailment supported by ontology languages either directly

[4] Even though in OntoWeb, due to the unavailability of the solution developed in this article, the problem was approached by caching the Web pages through a proxy server.

[5] The underlying rule language used for our approach is Datalog with stratified negation.

using Logic Programming techniques, e.g. [4, 25], or by relying on (available) logic databases[6] [22, 27]. Furthermore, a large expressive fragment of the recently standardized Web Ontology Language (OWL) can be implemented in logic databases [34].

2.1 Axiomatization of Ontology Languages

Systems like SilRi [7], CWM[7], Euler [25], JTP[8] or Triple [27] and ConceptBase [15] implement the semantics of a particular ontology language via a static axiomatization, i.e. a set of rules. For example, Figure 2 presents the Datalog axiomatization of the RDF vocabulary description language (RDFS) [5]. This axiomatization implements the semantics of RDF specified by the RDF model theory [13] (without datatype entailments and support for stronger iff semantics of domain and ranges). The ontology and associated data is stored in a single ternary predicate *t*, i.e. the extension of *t* stores all triples that constitute a particular RDF graph.

t(P,a,rdf:Property)	:- t(S,P,O).	*rdf1*
t(S,a,C)	:- t(P,domain,C), t(S,P,O).	*rdfs2*
t(O,a,C)	:- t(P,range,C), t(S,P,O).	*rdfs3*
t(S,a,Resource)	:- t(S,P,O).	*rdfs4a*
t(O,a,Resource)	:- t(S,P,O).	*rdfs4b*
t(P,subPropertyOf,R)	:- t(Q,subPropertyOf,R), t(P,subPropertyOf,Q).	*rdfs5a*
t(S,R,0)	:- t(P,subPropertyOf,R), t(S,P,O).	*rdfs6*
t(C,a,Class)	:- t(C,subClassOf,Resource).	*rdfs7*
t(A,subClassOf,C)	:- t(B,subClassOf,C), t(A,subClassOf,B).	*rdfs8*
t(S,a;B)	:- t(S,a,A), t(A,subClassOf,B).	*rdfs9*
t(X,subPropertyOf,member)	:- t(X,a,ContainerMembershipProperty).	*rdfs10*
t(X,subClassOf,Literal)	:- t(X,a,Datatype).	*rdfs11*
t(Resource,subClassOf,Y)	:- t(X,domain,Y), t(rdf:type,subPropertyOf,X).	*rdfs12*

Fig. 2. Static Datalog rules for implementing RDF(S)

2.2 Dynamic Rule Sets

The set of rules is typically not immutable. With the advent of higher layers of the Semantic Web stack, i.e. the rule layer, users can create their own rules.

[6] We use the term logic database over the older term deductive databases since the later is very closely associated with Datalog, a particular Logic Programming language that is frequently used in logic databases. Modern logic databases such as XSB [26] and CORAL [24] support more expressive Logic Programming languages that include function symbols and nested expressions. Furthermore, several lectures, e.g. http://user.it.uu.se/˜voronkorov/ddb.htm nowadays use this term.

[7] `http://www.w3.org/2000/10/swap/doc/cwm`

[8] `http://ksl.stanford.edu/software/jtp/`