Stéphane Bressan · Stefano Ceri
Ela Hunt · Zachary G. Ives
Zohra Bellahsène · Michael Rys
Rainer Unland (Eds.)

LNCS 3671

Database and XML Technologies

**Third International XML Database Symposium, XSym 2005
Trondheim, Norway, August 2005
Proceedings**

Springer

Stéphane Bressan Stefano Ceri
Ela Hunt Zachary G. Ives
Zohra Bellahsène Michael Rys
Rainer Unland (Eds.)

Database and XML Technologies

Third International XML Database Symposium, XSym 2005
Trondheim, Norway, August 28-29, 2005
Proceedings

Springer

Volume Editors

Stéphane Bressan
National University of Singapore, Department of Computer Science
School of Computing
3 Science drive 2, 117543 Singapore, Republic of Singapore
E-mail: steph@nus.edu.sg

Stefano Ceri
Politecnico di Milano, Dipartimento di Elettronica e Informazione
Via Ponzio, 34/5, 20133 Milano, Italy
E-mail: ceri@elet.polimi.it

Ela Hunt
University of Glasgow, Department of Computing Science
Lilybank Gardens 8-17, Glasgow G12 8QQ, UK
E-mail: ela@dcs.gla.ac.uk

Zachary G. Ives
University of Pennsylvania, Computer and Information Science Department
3330 Walnut Street, Philadelphia, PA 19104-6389, USA
E-mail: zives@atcis.upenn.edu

Zohra Bellahsène
LIRMM UMR 5506 CNRS/Université Montpellier II
161 Rue Ada, 34392 Montpellier, France
E-mail: bella@lirmm.fr

Michael Rys
Microsoft Corporation, One Microsoft Way, Redmond, WA 98052, USA
E-mail: mrys@mircosoft.com

Rainer Unland
University of Duisburg-Essen
Institute for Computer Science and Business Information Systems
Schützenbahn 70, 45117 Essen, Germany
E-mail: UnlandR@informatik.uni-essen.de

Library of Congress Control Number: 2005931472
CR Subject Classification (1998): H.2, H.3, H.4, C.2.4

ISSN 0302-9743
ISBN-10 3-540-28583-0 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-28583-0 Springer Berlin Heidelberg New York

Printed in Germany
Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 11547273 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3671

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

Editorial Board

Preface

This year marks an exciting time in the XML-database space: XQuery is moving closer to becoming a full W3C Recommendation, and the "Big 3" database vendors (IBM, Oracle, Microsoft) are expected to release XQuery support in their relational DBMSs, joining a number of existing open source and commercial products. Thus, we are very pleased to feature an industrial paper (describing the XML-specific features of Microsoft SQL Server) as well as 14 research papers. XSym's focus this year was on building XML repositories, and papers discussed the following topics: indexing support for the evaluation of XPath and XQuery; benchmarks and algorithms for XQuery and XPath evaluation; algorithms for constraint satisfaction checking, information extraction, and subtree matching; and applications of XML in information systems.

This year, XSym also coordinated its efforts with the Database and Programming Languages Symposium, DBPL 2005. The resulting program included not only presentations of the papers in this proceedings, but also a joint DBPL-XSym keynote talk by Giuseppe Castagna, developer of the CDuce language for XML processing, and a joint panel on open XML research problems and challenges.

The organizers would like to express their gratitude to the XSym Program Committee and external reviewers for their efforts in providing very thorough evaluations of the submitted papers under significant time constraints and to Microsoft for their sponsorship and for the use of the Microsoft Conference Management Toolkit. We would also like to thank Gavin Bierman and Christoph Koch, the organizers of DBPL, for their efforts and their willingness to coordinate with us.

These proceedings are dedicated to Alberto Mendelzon who sadly passed away this year. As a strong supporter of and active contributor to this symposium series he will always remain in our memory.

Singapore, Milan, Montpellier, Glasgow
Philadelphia, Essen, Redmond

July 2005

Stéphane Bressan
Stefano Ceri
Zohra Bellahsene
Ela Hunt
Zachary Ives
Rainer Unland
Michael Rys

General Chair

Stéphane Bressan, National University of Singapore (Singapore)

General Co-chair

Stefano Ceri, Politecnico di Milano (Italy)

Organizing Chair

Zohra Bellahsene, LIRMM (France)

Program Committee Chairs

Ela Hunt, University of Glasgow (UK)
Zachary Ives, University of Pennsylvania (USA)

Proceedings

Rainer Unland, University of Duisburg-Essen (Germany)

Sponsorship

Michael Rys, Microsoft (USA)

Communications

Akmal B. Chaudhri, IBM ISV & Developer Relations (USA)

International Programme Committee

Ashraf Aboulnaga, University of Waterloo (Canada)
Sihem Amer-Yahia, AT&T Research (USA)
Ricardo Baeza-Yates, Universidad de Chile (Chile)
Veronique Benzaken, LRI – Université Paris XI (France)
Tiziana Catarci, University of Roma "La Sapienza" (Italy)
Yi Chen, University of Pennsylvania (USA)
Giovanna Guerrini, Università di Pisa (Italy)
Ashish Kumar Gupta, University of Washington (USA)
Raghav Kaushik, Microsoft Research (USA)
Qiong Luo, Hong Kong University of Science and Technology (China)
Ioana Manolescu, INRIA (France)
Peter McBrien, Imperial College London (UK)
Guido Moerkotte, University of Mannheim (Germany)
Felix Naumann, Humboldt University Berlin (Germany)
Werner Nutt, Heriot-Watt University (UK)
Beng Chin Ooi, National University of Singapore (Singapore)
M. Tamer Ozsu, University of Waterloo (Canada)
Tadeusz Pankowski, Poznan University of Technology (Poland)
Alexandra Poulovassilis, Birkbeck College, University of London (UK)
Prakash Ramanan, Wichita State University (USA)
Elke A. Rundensteiner, Worcester Polytechnic Institute (USA)
Arnaud Sahuguet, Bell Laboratories – Lucent Technologies (USA)
Monica Scannapieco, University of Roma "La Sapienza" (Italy)
Jayavel Shanmugasundaram, Cornell University (USA)
Jerome Simeon, IBM Research (USA)
Wang-Chiew Tan, University of California, Santa Cruz (USA)
Yannis Velegrakis, AT&T Research (USA)
Stratis Viglas, University of Edinburgh (UK)
Peter Wood, Birkbeck College, University of London (UK)
Yuqing Melanie Wu, Indiana University (USA)
Jun Yang, Duke University (USA)
Jeffrey Xu Yu, Chinese University of Hong Kong (China)

External Reviewers

Andrei Arion	LRI – Université Paris XI (France)
Patrick Bosc	LRI – Université Paris XI (France)
Chavdar Botev	Cornell University (USA)
Giuseppe Castagna	LRI – Université Paris XI (France)
Laura Chiticariu	University of California, Santa Cruz (USA)
David DeHaan	University of Waterloo (Canada)
Maged El-Sayed	Worcester Polytechnic Institute (USA)
Fan Yang	Cornell University (USA)
Mirian Halfeld-Ferrari	LRI – Université Paris XI (France)
Ming Jiang	Worcester Polytechnic Institute (USA)
Ming Lee	Worcester Polytechnic Institute (USA)
Diego Milano	University of Roma (Italy)
	University of Edinburgh (UK)
Feng Shao	Cornell University (USA)
Frank Tompa	University of Waterloo (Canada)
Song Wang	Worcester Polytechnic Institute (USA)
Rui Yang	National University of Singapore (Singapore)
Ning Zhang	University of Waterloo (Canada)

Lecture Notes in Computer Science

For information about Vols. 1–3548

please contact your bookseller or Springer

Vol. 3671: S. Bressan, S. Ceri, E. Hunt, Z.G. Ives, Z. Bellahsène, M. Rys, R. Unland (Eds.), Database and XML Technologies. X, 239 pages. 2005.

Vol. 3662: C. Baral, G. Greco, N. Leone, G. Terracina (Eds.), Logic Programming and Nonmonotonic Reasoning. XIII, 454 pages. 2005. (Subseries LNAI).

Vol. 3659: J.R. Rao, B. Sunar (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2005. XIV, 458 pages. 2005.

Vol. 3654: S. Jajodia, D. Wijesekera (Eds.), Data and Applications Security XIX. X, 353 pages. 2005.

Vol. 3653: M. Abadi, L.d. Alfaro (Eds.), CONCUR 2005 – Concurrency Theory. XIV, 578 pages. 2005.

Vol. 3649: W.M.P. van der Aalst, B. Benatallah, F. Casati, F. Curbera (Eds.), Business Process Management. XII, 472 pages. 2005.

Vol. 3645: D.-S. Huang, X.-P. Zhang, G.-B. Huang (Eds.), Advances in Intelligent Computing, Part II. XIII, 1010 pages. 2005.

Vol. 3644: D.-S. Huang, X.-P. Zhang, G.-B. Huang (Eds.), Advances in Intelligent Computing, Part I. XXVII, 1101 pages. 2005.

Vol. 3639: P. Godefroid (Ed.), Model Checking Software. XI, 289 pages. 2005.

Vol. 3638: A. Butz, B. Fisher, A. Krüger, P. Olivier (Eds.), Smart Graphics. XI, 269 pages. 2005.

Vol. 3636: M.J. Blesa, C. Blum, A. Roli, M. Sampels (Eds.), Hybrid Metaheuristics. XII, 155 pages. 2005.

Vol. 3634: L. Ong (Ed.), Computer Science Logic. XI, 567 pages. 2005.

Vol. 3633: C. Bauzer Medeiros, M. Egenhofer, E. Bertino (Eds.), Advances in Spatial and Temporal Databases. XIII, 433 pages. 2005.

Vol. 3632: R. Nieuwenhuis (Ed.), Automated Deduction – CADE-20. XIII, 459 pages. 2005. (Subseries LNAI).

Vol. 3629: J.L. Fiadeiro, N. Harman, M. Roggenbach, J. Rutten (Eds.), Algebra and Coalgebra in Computer Science. XI, 457 pages. 2005.

Vol. 3627: C. Jacob, M.L. Pilat, P.J. Bentley, J. Timmis (Eds.), Artificial Immune Systems. XII, 500 pages. 2005.

Vol. 3626: B. Ganter, G. Stumme, R. Wille (Eds.), Formal Concept Analysis. X, 349 pages. 2005. (Subseries LNAI).

Vol. 3625: S. Kramer, B. Pfahringer (Eds.), Inductive Logic Programming. XIII, 427 pages. 2005. (Subseries LNAI).

Vol. 3624: C. Chekuri, K. Jansen, J.D.P. Rolim, L. Trevisan (Eds.), Approximation, Randomization and Combinatorial Optimization. XI, 495 pages. 2005.

Vol. 3623: M. Liśkiewicz, R. Reischuk (Eds.), Fundamentals of Computation Theory. XV, 576 pages. 2005.

Vol. 3621: V. Shoup (Ed.), Advances in Cryptology – CRYPTO 2005. XI, 568 pages. 2005.

Vol. 3620: H. Muñoz-Avila, F. Ricci (Eds.), Case-Based Reasoning Research and Development. XV, 654 pages. 2005. (Subseries LNAI).

Vol. 3619: X. Lu, W. Zhao (Eds.), Networking and Mobile Computing. XXIV, 1299 pages. 2005.

Vol. 3615: B. Ludäscher, L. Raschid (Eds.), Data Integration in the Life Sciences. XII, 344 pages. 2005. (Subseries LNBI).

Vol. 3614: L. Wang, Y. Jin (Eds.), Fuzzy Systems and Knowledge Discovery, Part II. XLI, 1314 pages. 2005. (Subseries LNAI).

Vol. 3613: L. Wang, Y. Jin (Eds.), Fuzzy Systems and Knowledge Discovery, Part I. XLI, 1334 pages. 2005. (Subseries LNAI).

Vol. 3612: L. Wang, K. Chen, Y. S. Ong (Eds.), Advances in Natural Computation, Part III. LXI, 1326 pages. 2005.

Vol. 3611: L. Wang, K. Chen, Y. S. Ong (Eds.), Advances in Natural Computation, Part II. LXI, 1292 pages. 2005.

Vol. 3610: L. Wang, K. Chen, Y. S. Ong (Eds.), Advances in Natural Computation, Part I. LXI, 1302 pages. 2005.

Vol. 3608: F. Dehne, A. López-Ortiz, J.-R. Sack (Eds.), Algorithms and Data Structures. XIV, 446 pages. 2005.

Vol. 3607: J.-D. Zucker, L. Saitta (Eds.), Abstraction, Reformulation and Approximation. XII, 376 pages. 2005. (Subseries LNAI).

Vol. 3606: V. Malyshkin (Ed.), Parallel Computing Technologies. XII, 470 pages. 2005.

Vol. 3604: R. Martin, H. Bez, M. Sabin (Eds.), Mathematics of Surfaces XI. IX, 473 pages. 2005.

Vol. 3603: J. Hurd, T. Melham (Eds.), Theorem Proving in Higher Order Logics. IX, 409 pages. 2005.

Vol. 3602: R. Eigenmann, Z. Li, S.P. Midkiff (Eds.), Languages and Compilers for High Performance Computing. IX, 486 pages. 2005.

Vol. 3599: U. Aßmann, M. Aksit, A. Rensink (Eds.), Model Driven Architecture. X, 235 pages. 2005.

Vol. 3598: H. Murakami, H. Nakashima, H. Tokuda, M. Yasumura, Ubiquitous Computing Systems. XIII, 275 pages. 2005.

Vol. 3597: S. Shimojo, S. Ichii, T.W. Ling, K.-H. Song (Eds.), Web and Communication Technologies and Internet-Related Social Issues - HSI 2005. XIX, 368 pages. 2005.

Vol. 3596: F. Dau, M.-L. Mugnier, G. Stumme (Eds.), Conceptual Structures: Common Semantics for Sharing Knowledge. XI, 467 pages. 2005. (Subseries LNAI).

Vol. 3595: L. Wang (Ed.), Computing and Combinatorics. XVI, 995 pages. 2005.

Vol. 3594: J.C. Setubal, S. Verjovski-Almeida (Eds.), Advances in Bioinformatics and Computational Biology. XIV, 258 pages. 2005. (Subseries LNBI).

Vol. 3593: V. Mařík, R. W. Brennan, M. Pěchouček (Eds.), Holonic and Multi-Agent Systems for Manufacturing. XI, 269 pages. 2005. (Subseries LNAI).

Vol. 3592: S. Katsikas, J. Lopez, G. Pernul (Eds.), Trust, Privacy and Security in Digital Business. XII, 332 pages. 2005.

Vol. 3591: M.A. Wimmer, R. Traunmüller, Å. Grönlund, K.V. Andersen (Eds.), Electronic Government. XIII, 317 pages. 2005.

Vol. 3590: K. Bauknecht, B. Pröll, H. Werthner (Eds.), E-Commerce and Web Technologies. XIV, 380 pages. 2005.

Vol. 3589: A M. Tjoa, J. Trujillo (Eds.), Data Warehousing and Knowledge Discovery. XVI, 538 pages. 2005.

Vol. 3588: K.V. Andersen, J. Debenham, R. Wagner (Eds.), Database and Expert Systems Applications. XX, 955 pages. 2005.

Vol. 3587: P. Perner, A. Imiya (Eds.), Machine Learning and Data Mining in Pattern Recognition. XVII, 695 pages. 2005. (Subseries LNAI).

Vol. 3586: A.P. Black (Ed.), ECOOP 2005 - Object-Oriented Programming. XVII, 631 pages. 2005.

Vol. 3584: X. Li, S. Wang, Z.Y. Dong (Eds.), Advanced Data Mining and Applications. XIX, 835 pages. 2005. (Subseries LNAI).

Vol. 3583: R.W. H. Lau, Q. Li, R. Cheung, W. Liu (Eds.), Advances in Web-Based Learning – ICWL 2005. XIV, 420 pages. 2005.

Vol. 3582: J. Fitzgerald, I.J. Hayes, A. Tarlecki (Eds.), FM 2005: Formal Methods. XIV, 558 pages. 2005.

Vol. 3581: S. Miksch, J. Hunter, E. Keravnou (Eds.), Artificial Intelligence in Medicine. XVII, 547 pages. 2005. (Subseries LNAI).

Vol. 3580: L. Caires, G.F. Italiano, L. Monteiro, C. Palamidessi, M. Yung (Eds.), Automata, Languages and Programming. XXV, 1477 pages. 2005.

Vol. 3579: D. Lowe, M. Gaedke (Eds.), Web Engineering. XXII, 633 pages. 2005.

Vol. 3578: M. Gallagher, J. Hogan, F. Maire (Eds.), Intelligent Data Engineering and Automated Learning - IDEAL 2005. XVI, 599 pages. 2005.

Vol. 3577: R. Falcone, S. Barber, J. Sabater-Mir, M.P. Singh (Eds.), Trusting Agents for Trusting Electronic Societies. VIII, 235 pages. 2005. (Subseries LNAI).

Vol. 3576: K. Etessami, S.K. Rajamani (Eds.), Computer Aided Verification. XV, 564 pages. 2005.

Vol. 3575: S. Wermter, G. Palm, M. Elshaw (Eds.), Biomimetic Neural Learning for Intelligent Robots. IX, 383 pages. 2005. (Subseries LNAI).

Vol. 3574: C. Boyd, J.M. González Nieto (Eds.), Information Security and Privacy. XIII, 586 pages. 2005.

Vol. 3573: S. Etalle (Ed.), Logic Based Program Synthesis and Transformation. VIII, 279 pages. 2005.

Vol. 3572: C. De Felice, A. Restivo (Eds.), Developments in Language Theory. XI, 409 pages. 2005.

Vol. 3571: L. Godo (Ed.), Symbolic and Quantitative Approaches to Reasoning with Uncertainty. XVI, 1028 pages. 2005. (Subseries LNAI).

Vol. 3570: A. S. Patrick, M. Yung (Eds.), Financial Cryptography and Data Security. XII, 376 pages. 2005.

Vol. 3569: F. Bacchus, T. Walsh (Eds.), Theory and Applications of Satisfiability Testing. XII, 492 pages. 2005.

Vol. 3568: W.-K. Leow, M.S. Lew, T.-S. Chua, W.-Y. Ma, L. Chaisorn, E.M. Bakker (Eds.), Image and Video Retrieval. XVII, 672 pages. 2005.

Vol. 3567: M. Jackson, D. Nelson, S. Stirk (Eds.), Database: Enterprise, Skills and Innovation. XII, 185 pages. 2005.

Vol. 3566: J.-P. Banâtre, P. Fradet, J.-L. Giavitto, O. Michel (Eds.), Unconventional Programming Paradigms. XI, 367 pages. 2005.

Vol. 3565: G.E. Christensen, M. Sonka (Eds.), Information Processing in Medical Imaging. XXI, 777 pages. 2005.

Vol. 3564: N. Eisinger, J. Małuszyński (Eds.), Reasoning Web. IX, 319 pages. 2005.

Vol. 3562: J. Mira, J.R. Álvarez (Eds.), Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach, Part II. XXIV, 636 pages. 2005.

Vol. 3561: J. Mira, J.R. Álvarez (Eds.), Mechanisms, Symbols, and Models Underlying Cognition, Part I. XXIV, 532 pages. 2005.

Vol. 3560: V.K. Prasanna, S. Iyengar, P.G. Spirakis, M. Welsh (Eds.), Distributed Computing in Sensor Systems. XV, 423 pages. 2005.

Vol. 3559: P. Auer, R. Meir (Eds.), Learning Theory. XI, 692 pages. 2005. (Subseries LNAI).

Vol. 3558: V. Torra, Y. Narukawa, S. Miyamoto (Eds.), Modeling Decisions for Artificial Intelligence. XII, 470 pages. 2005. (Subseries LNAI).

Vol. 3557: H. Gilbert, H. Handschuh (Eds.), Fast Software Encryption. XI, 443 pages. 2005.

Vol. 3556: H. Baumeister, M. Marchesi, M. Holcombe (Eds.), Extreme Programming and Agile Processes in Software Engineering. XIV, 332 pages. 2005.

Vol. 3555: T. Vardanega, A.J. Wellings (Eds.), Reliable Software Technology – Ada-Europe 2005. XV, 273 pages. 2005.

Vol. 3554: A. Dey, B. Kokinov, D. Leake, R. Turner (Eds.), Modeling and Using Context. XIV, 572 pages. 2005. (Subseries LNAI).

Vol. 3553: T.D. Hämäläinen, A.D. Pimentel, J. Takala, S. Vassiliadis (Eds.), Embedded Computer Systems: Architectures, Modeling, and Simulation. XV, 476 pages. 2005.

Vol. 3552: H. de Meer, N. Bhatti (Eds.), Quality of Service – IWQoS 2005. XVIII, 400 pages. 2005.

Vol. 3551: T. Härder, W. Lehner (Eds.), Data Management in a Connected World. XIX, 371 pages. 2005.

Table of Contents

Documents and Biometrical Applications

Industrial Session

Panel (Together with DBPL)

Patterns and Types for Querying XML Documents *

Giuseppe Castagna

CNRS, École Normale Supérieure de Paris, France

In order to manipulate XML data, a programming or query language should provide some primitives to deconstruct them, in particular to pinpoint and capture some subparts of the data.

Among various proposals for primitives for deconstructing XML data, two different and complementary approaches seem to clearly stem from practise: path expressions (usually XPath paths [7], but also the "dot" navigation of Cω [3]) and regular expression patterns [13].

Path expressions are navigational primitives that point out where to capture data substructures. They (and those of Cω, in particular) closely resemble the homonymous primitives used by OQL [9] in the contexts of OODB query languages with the difference that instead of sets of objects they return sets or sequences of elements: more precisely all elements that can be reached following the path at issue. These primitives are at the basis of standard languages such as XSLT [8] or XQuery [4].

More recently, a new kind of deconstructing primitives was proposed, regular expression patterns [13], which extend by regular expressions the pattern matching primitive as popularised by functional languages such as ML and Haskell. Regular expression patterns were first introduced in the XDuce [12] programming language and are becoming more and more popular, since they are being adopted by such quite different languages as CDuce [1] (a general purpose extension of the XDuce language) and its query language CQL [2], Xtatic [10] (an extension of C#), Scala [15] (a general purpose Java-like object-oriented language that compiles into Java bytecode), XHaskell [14] as well as the extension of Haskell proposed in [5].

The two kinds of primitives are not antagonists, but rather orthogonal and complementary. Path expressions implement a "vertical" exploration of data as they capture elements that may be at different depths, while patterns perform a "horizontal" exploration of data since they are able to perform finer grained decomposition on sequences of elements. The two kinds of primitives are quite useful and they mutually complement nicely. Therefore, it would seem natural to integrate both of them in a query or programming language for XML. Despite of that, we are aware of just two works in which both primitives are embedded (and, yet, loosely coupled): in CQL it is possible to write select-from-where expressions, where regular expression patterns are applied in the from clause to sequences that are returned by XPath-like expressions; Gapeyev and Pierce [11] show how it is possible to use regular expression patterns with an all match semantics to encode a subset of XPath and plan to use this encoding to add XPath to the Xtatic programming language.

The reason for the lack of study of the integration of these two primitives may be due to the fact that each of them is adopted by a different community: regular patterns

* Joint talk with DBPL 2005. Full version available in the Proc. of the *10th Intl. Symp. on Database Programming Languages*, G. Bierman and C. Koch eds., LNCS, Springer, 2005.

S. Bressan et al. (Eds.): XSym 2005, LNCS 3671, pp. 1–3, 2005.

are almost confined to the programming language community while XPath expressions are pervasive in the database community.

The goal of this lecture is to give a brief presentation of the regular pattern expressions style together with the type system to which they are tightly connected, that is the semantic subtyping based type systems [6]. We are not promoting the use of these to the detriment of path expressions, since we think that the two approaches should be integrated in the same language and we see in that a great opportunity of collaboration between the database and the programming languages communities. Since the author belongs to latter, this lecture tries to describe the pattern approach addressing some points that should be of interest to the database community as well. In particular, after a general overview of regular expression patterns and types in which we show how to embed patterns in a select·from·where expression, we discuss several usages of these patterns/types, going from the classic use for partial correctness and schema specification to the definition of new data iterators, from the specification of efficient run-time to the definition of logical pattern-specific query optimisations.

References

1. V. Benzaken, G. Castagna, and A. Frisch. CDuce: an XML-friendly general purpose language. In *ICFP '03, 8th ACM International Conference on Functional Programming*, pages 51–63, Uppsala, Sweden, 2003. ACM Press.
2. V. Benzaken, G. Castagna, and C. Miachon. A full pattern-based paradigm for XML query processing. In *PADL 05, 7th International Symposium on Practical Aspects of Declarative Languages*, number 3350 in LNCS, pages 235–252. Springer, January 2005.
3. Gavin Bierman, Erik Meijer, and Wolfram Schulte. The essence of data access in Cw. In *Proc. of ECOOP '2005, European Conference on Object-Oriented Programming*, volume 3586 of *Lecture Notes in Computer Science*. Springer, 2005.
4. S. Boag, D. Chamberlin, M. Fernandez, D. Florescu, J. Robie, J. Siméon, and M. Stefanescu. *XQuery 1.0: An XML Query Language*. W3C Working Draft, `http://www.w3.org/TR/xquery/`, May 2003.
5. Niklas Broberg, Andreas Farre, and Josef Svenningsson. Regular expression patterns. In *ICFP '04: Proceedings of the ninth ACM SIGPLAN international conference on Functional programming*, pages 67–78, New York, NY, USA, 2004. ACM Press.
6. G. Castagna and A. Frisch. A gentle introduction to semantic subtyping. In Proceedings of *PPDP '05, the 7th ACM SIGPLAN International Symposium on Principles and Practice of Declarative Programming*, ACM Press (full paper) and *ICALP '05, 32nd International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science n. 3580, Springer (short abstract), Lisboa, Portugal, 2005. Joint ICALP-PPDP keynote talk.
7. J. Clark and S. DeRose. *XML Path Language (XPath)*. W3C Recommendation, `http://www.w3.org/TR/xpath/`, November 1999.
8. James Clark. *XSL Transformations (XSLT)*. W3C Recommendation, `http://www.w3.org/TR/xslt/`, November 1999.
9. Sophie Cluet. Designing OQL: allowing objects to be queried. *Inf. Syst.*, 23(5):279–305, 1998.
10. Vladimir Gapeyev and Benjamin C. Pierce. Regular object types. In *European Conference on Object-Oriented Programming (ECOOP), Darmstadt, Germany*, 2003. A preliminary version was presented at FOOL '03.

11. Vladimir Gapeyev and Benjamin C. Pierce. Paths into patterns. Technical Report MS-CIS-04-25, University of Pennsylvania, October 2004.
12. Haruo Hosoya and Benjamin C. Pierce. XDuce: A typed XML processing language. In *Proceedings of Third International Workshop on the Web and Databases (WebDB2000)*, 2000.
13. Haruo Hosoya and Benjamin C. Pierce. Regular expression pattern matching for XML. In *The 25th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, 2001.
14. K. Zhuo Ming Lu and M. Sulzmann. An implementation of subtyping among regular expression types. In *Proc. of APLAS'04*, volume 3302 of LNCS, pages 57–73. Springer-Verlag, 2004.
15. Martin Odersky, Philippe Altherr, Vincent Cremet, Burak Emir, Sebastian Maneth, Stéphane Micheloud, Nikolay Mihaylov, Michel Schinz, Erik Stenman, and Matthias Zenger. An overview of the scala programming language. Technical Report IC/2004/64, École Polytechnique Fédérale de Lausanne, 2004. Latest version at `http://scala.epfl.ch`.

Checking Functional Dependency Satisfaction in XML

Millist W. Vincent and Jixue Liu

School of Computer and Information Science
University of South Australia
{millist.vincent,jixue.liu}@unisa.edu.au

Abstract. Recently, the issue of functional dependencies in XML (XFDs) have been investigated. In this paper we consider the problem of checking the satisfaction of an XFD in an XML document. We present an efficient algorithm for the problem that is linear in the size of the XML document and linear in the number of XFDs to be checked. Also, our technique can be easily extended to efficiently incrementally check XFD satisfaction.

1 Introduction

The eXtensible Markup Language (XML) [5] has recently emerged as a standard for data representation and interchange on the Internet. While providing syntactic flexibility, XML provides little semantic content and as a result several papers have addressed the topic of how to improve the semantic expressiveness of XML. Among the most important of these approaches has been that of defining *integrity constraints* in XML [7]. Several different classes of integrity constraints for XML have been defined including key constraints [6], path constraints [8], and inclusion constraints [10, 11] and properties such as axiomatization and satisfiability have been investigated for these constraints. However, one topic that has been identified as an open problem in XML research [16] and which has been little investigated is how to extend the oldest and most well studied integrity constraint in relational databases, namely a *functional dependency* (FD), to XML and then how to develop a normalization theory for XML. This problem is not of just theoretical interest. The theory of FDs and normalization forms the cornerstone of practical relational database design and the development of a similar theory for XML will similarly lay the foundation for understanding how to design XML documents.

Recently, two approaches have been given for defining functional dependencies in XML (called XFDs). The first [1–3], proposed a definition based on the notion of a 'tree tuple' which in turn is based on the total unnesting of a relation [4]. More recently, we have proposed an alternative 'closest node' definition [14], which is based on paths and path instances that has similarity with the approach in [6] to defining keys in XML. This relationship between keys as defined in [6] and XFDs as defined in [14] extends further, as it was shown in [14] that in the

S. Bressan et al. (Eds.): XSym 2005, LNCS 3671, pp. 4–17, 2005.

case of simple paths, keys in XML are a special case of XFDs in the same way that keys in relational databases are a special case of FDs.

In general, the two approaches to defining XFDs are not comparable since they treat missing information in the XML document differently and the approach in [1–3] assumes the existence of a DTD whereas the approach in [14] does not. Howevever, we have recently shown that [15], in spite of the very different approaches used in [1–3] and [14], the two aproaches coincide for a large class of XML documents. In particular, we have shown that the definitions coincide for XML documents with no missing information conforming to a nonrecursve, disjunction free DTD. This class includes XML documents derived from complete relational databases using any 'non pathological' mapping. It has also been shown that in this situation, for mappings from a relation to an XML document defined by first mapping to a nested relation via an arbitrary sequence of nest and unnest operations, then followed my a direct mapping to XML, FDs in relations map to XFDs in XML. Hence there is a natural correspondence between FDs and XFDs.

In this paper we address the problem of developing an efficient algorithm for checking whether an XML document satisfies a set of XFDs as defined in [14]. We develop an algorithm which requires only one pass of the XML document and whose running time is linear in the size of the XML document and linear in the size of the number of XFDs. The algorithm uses an innovative method based on a multi level extension of extendible hashing. We also investigate the effect of the size on the number of paths on the l.h.s. of the XFD and show that the running time is both linear in the number of paths and also increases quite slowly with the number of paths.

Although the issue of developing checking the satisfaction of 'tree tuple' XFDs was not addressed in [1–3], testing satisfaction using the definitions in [1–3] directly is likely to be quite expensive. This is because there are three steps involved in the approach of [1–3]. The first is to generate a set of tuples from the total unnesting of an XML document. This set is likely to be much larger than the original XML document since unnesting generates all possible combinations amongst elements. The second step is to generate the set of tree tuples, since not all tuples generated from the total unnesting are 'tree tuples'. This is done by generating a special XML tree (document) from a tuple and checking if the document so generated is subsumed by the original XML tree (document). Once again this is likely to be an expensive procedure since it may require that the number of times the XML document is scanned is the same as the number of tuples in the total unnesting. In contrast, our method requires only one scan of the XML document. Finally, the definition in [1–3] requires scanning the set of tree tuples to check for satisfaction in a manner similar to ordinary FD satisfaction. This last step is common also to our approach.

The rest of this paper is organized as follows. Section 2 contains some preliminary definitions that we need before defining XFDs. We model an XML document as a tree as follows. In Section 3 the definition of an XFD is presented and the essential ideas of our algorithm are presented. Section 4 contains details