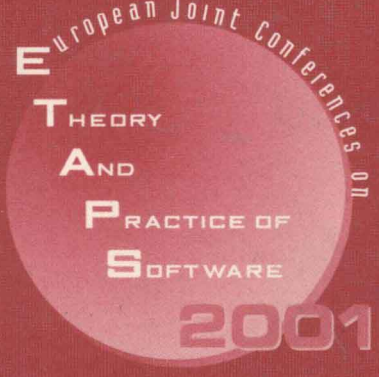
David Sands (Ed.)

LNCS 2028

Programming Languages and Systems

10th European Symposium on Programming, ESOP 2001
Held as Part of the Joint European Conferences
on Theory and Practice of Software, ETAPS 2001
Genova, Italy, April 2001, Proceedings



Springer

David Sands (Ed.)

Programming Languages and Systems

10th European Symposium on Programming, ESOP 2001
Held as Part of the Joint European Conferences
on Theory and Practice of Software, ETAPS 2001
Genova, Italy, April 2-6, 2001
Proceedings

Springer

Series Editors

Gerhard Goos, Karlsruhe University, Germany
Juris Hartmanis, Cornell University, NY, USA
Jan van Leeuwen, Utrecht University, The Netherlands

Volume Editor

David Sands
Chalmers University of Technology and Göteborg University
Department of Computing Science
412 96 Göteborg, Sweden
E-mail: dave@cs.chalmers.se

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Programming languages and systems : proceedings / 10th European Symposium on Programming, ESOP 2001, held as part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2001, Genova, Italy, April 2 - 6, 2001. David Sands (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Hong Kong ; London ; Milan ; Paris ; Singapore ; Tokyo : Springer, 2001
(Lecture notes in computer science ; Vol. 2028)
ISBN 3-540-41862-8

CR Subject Classification (1998): D.3, D.1-2, F.3-4, E.1

ISSN 0302-9743
ISBN 3-540-41862-8 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

http://www.springer.de

Printed in Germany

Typesetting: Camera-ready by author, data conversion by PTP-Berlin, Stefan Sossna
Printed on acid-free paper SPIN: 10782434 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 2028

Edited by G. Goos, J. Hartmanis and J. van Leeuwen

Foreword

ETAPS 2001 was the fourth instance of the European Joint Conferences on Theory and Practice of Software. ETAPS is an annual federated conference that was established in 1998 by combining a number of existing and new conferences. This year it comprised five conferences (FOSSACS, FASE, ESOP, CC, TACAS), ten satellite workshops (CMCS, ETI Day, JOSES, LDTA, MMAABS, PFM, RelMiS, UNIGRA, WADT, WTUML), seven invited lectures, a debate, and ten tutorials.

The events that comprise ETAPS address various aspects of the system development process, including specification, design, implementation, analysis, and improvement. The languages, methodologies, and tools which support these activities are all well within its scope. Different blends of theory and practice are represented, with an inclination towards theory with a practical motivation on one hand and soundly-based practice on the other. Many of the issues involved in software design apply to systems in general, including hardware systems, and the emphasis on software is not intended to be exclusive.

ETAPS is a loose confederation in which each event retains its own identity, with a separate program committee and independent proceedings. Its format is open-ended, allowing it to grow and evolve as time goes by. Contributed talks and system demonstrations are in synchronized parallel sessions, with invited lectures in plenary sessions. Two of the invited lectures are reserved for "unifying" talks on topics of interest to the whole range of ETAPS attendees. The aim of cramming all this activity into a single one-week meeting is to create a strong magnet for academic and industrial researchers working on topics within its scope, giving them the opportunity to learn about research in related areas, and thereby to foster new and existing links between work in areas that were formerly addressed in separate meetings.

ETAPS 2001 was hosted by the Dipartimento di Informatica e Scienze dell'Informazione (DISI) of the Università di Genova and was organized by the following team:

Egidio Astesiano (General Chair)
Eugenio Moggi (Organization Chair)
Maura Cerioli (Satellite Events Chair)
Gianna Reggio (Publicity Chair)
Davide Ancona
Giorgio Delzanno
Maurizio Martelli

with the assistance of Convention Bureau Genova. Tutorials were organized by Bernhard Rumpe (TU München). Overall planning for ETAPS conferences is the responsibility of the ETAPS Steering Committee, whose current membership is:

Egidio Astesiano (Genova), Ed Brinksma (Enschede), Pierpaolo Degano (Pisa), Hartmut Ehrig (Berlin), José Fiadeiro (Lisbon), Marie-Claude Gaudel (Paris), Susanne Graf (Grenoble), Furio Honsell (Udine), Nigel Horspool (Victoria), Heinrich Hußmann (Dresden), Paul Klint (Amsterdam), Daniel Le Métayer (Rennes), Tom Maibaum (London), Tiziana Margaria (Dortmund), Ugo Montanari (Pisa), Mogens Nielsen (Aarhus), Hanne Riis Nielson (Aarhus), Fernando Orejas (Barcelona), Andreas Podelski (Saarbrücken), David Sands (Göteborg), Don Sannella (Edinburgh), Perdita Stevens (Edinburgh), Jerzy Tiuryn (Warsaw), David Watt (Glasgow), Herbert Weber (Berlin), Reinhard Wilhelm (Saarbrücken)

ETAPS 2001 was organized in cooperation with

the Association for Computing Machinery
the European Association for Programming Languages and Systems
the European Association of Software Science and Technology
the European Association for Theoretical Computer Science

and received generous sponsorship from:

ELSAG
Fondazione Cassa di Risparmio di Genova e Imperia
INDAM - Gruppo Nazionale per l'Informatica Matematica (GNIM)
Marconi
Microsoft Research
Telecom Italia
TXT e-solutions
Università di Genova

I would like to express my sincere gratitude to all of these people and organizations, the program committee chairs and PC members of the ETAPS conferences, the organizers of the satellite events, the speakers themselves, and finally Springer-Verlag for agreeing to publish the ETAPS proceedings.

January 2001

Donald Sannella
ETAPS Steering Committee chairman

Preface

This volume contains the 28 papers presented at ESOP 2001, the Tenth European Symposium on Programming, which took place in Genova, Italy, April 4–6, 2001. The ESOP series began in 1986, and addresses both practical and theoretical issues in the design, specification, and analysis of programming languages and systems.

The call for ESOP 2001 encouraged papers addressing (but not limited to)

- Programming paradigms (including functional, logic, concurrent, and object-oriented) and their integration;
- Semantics with applications to the development of correct, secure, and efficient software and systems;
- Advanced type systems, program analysis, program transformation.

The volume begins with two invited contributions. The first contribution belongs to ETAPS as a whole, and accompanies the "unifying" ETAPS invited talk given by Luca Cardelli. The second contribution is from the ESOP invited speaker, John Mitchell. The remaining 26 papers were selected by the program committee from the 76 submissions, and include one short paper which accompanied a tool-demo presentation.

Each submission was reviewed by at least three referees, and papers were selected in the latter stages of a two week discussion phase. My thanks to the members of the program committee and other referees for their hard work. Thanks also to Christian Probst for help with the conference management software, and to Don Sannella for steering the ETAPS ship so smoothly.

January 2001 David Sands

Organization

Program Chair

David Sands	Chalmers and Göteborg University, Sweden

Program Committee

Martín Abadi	Bell Labs, USA
Radhia Cousot	CNRS and École Polytechnique, France
Mads Dam	KTH Kista, Sweden
Andrew D. Gordon	Microsoft Research, UK
Robert Harper	CMU Pittsburgh, USA
Nevin Heintze	Bell Labs, USA
Daniel Le Métayer	Trusted Logic, France
Florence Maraninchi	Grenoble I/Verimag, France
Catuscia Palamidessi	Penn State, USA
Mooly Sagiv	Tel-Aviv University, Israel
David Sands	Chalmers and Göteborg University, Sweden
Peter Sestoft	KVL and ITU Copenhagen, Denmark
Harald Søndergaard	The University of Melbourne, Australia

Additional Referees

Johan Agat
Karine Altisen
Pierre Berlioux
Bruno Blanchet
Valentin Bonnard
Glenn Bruns
Michele Bugliesi
Luca Cardelli
Giuseppe Castagna
Jan Cederquist
Thomas Colcombet
Seth Copen Goldstein
Agostino Cortesi
Patrick Cousot
Karl Crary
Olivier Danvy
Ewen Denney
Nachum Dershowitz
Nurit Dor
Tyson Dowd
Conal Elliot
Martin Elsman
Jérôme Feret
Cedric Fournet
Pascal Fradet
Nissim Francez
Lars-Åke Fredlund
Stephen Freund
Roberto Giacobazzi
Pabla Giambiagi
Kevin Glynn
Gregor Goessler
Orna Grumberg
Dilian Gurov
Jörgen Gustavsson
Thomas Hallgren
Gregoire Hamon
John Hannan
Fritz Henglein
Charles Hymans
Daniel Jackson
Thomas Jensen
Mark P. Jones
Simon Jones
Jan Jurjens
Per Kreuger
John Lamping
Cosimo Laneve
Julia Lawall
Peter Lee
Bjorn Lisper
Francesco Logozzo
Renaud Marlet
Andres Martinelli
Damien Massé
Laurent Mauborgne
Antoine Miné
David Monniaux
Laurent Mounier
Lee Naish
Xavier Nicollin
Thomas Noll
Martin Odersky
Richard O'Keefe
Dino Oliva
Catherine Oriat
Gordon Pace
Joachim Parrow
Simon Peyton Jones
Frank Pfenning
François Pottier
K. V. S. Prasad
Elisa Quintarelli
C.R. Ramakrishnan
Francesco Ranzato
Julian Rathke
Jakob Rehof
Jon Riecke
Hanne Riis Nielson
Claudio Russo
Andrei Sabelfeld
Francesca Scozzari
Ran Shaham
Vitaly Shmatikov
Zoltan Somogyi
Fausto Spoto
Peter J. Stuckey
Martin Sulzmann
Mario Südholt
Tommy Thorn
Frank Valencia
Bjorn Victor
Ramesh Viswanathan
Jan Vitek
Jose-Luis Vivas
David Walker
Eran Yahav
Amiram Yehudai
Gianluigi Zavattaro

Springer
Berlin
Heidelberg
New York
Barcelona
Hong Kong
London
Milan
Paris
Singapore
Tokyo

Lecture Notes in Computer Science

For information about Vols. 1–1931
please contact your bookseller or Springer-Verlag

Vol. 1812: J. Wyatt, J. Demiris (Eds.), Advances in Robot Learning. Proceedings, 1999. VII, 165 pages. 2000. (Subseries LNAI).

Vol. 1932: Z.W. Raś, S. Ohsuga (Eds.), Foundations of Intelligent Systems. Proceedings, 2000. XII, 646 pages. (Subseries LNAI).

Vol. 1933: R.W. Brause, E. Hanisch (Eds.), Medical Data Analysis. Proceedings, 2000. XI, 316 pages. 2000.

Vol. 1934: J.S. White (Ed.), Envisioning Machine Translation in the Information Future. Proceedings, 2000. XV, 254 pages. 2000. (Subseries LNAI).

Vol. 1935: S.L. Delp, A.M. DiGioia, B. Jaramaz (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2000. Proceedings, 2000. XXV, 1250 pages. 2000.

Vol. 1936: P. Robertson, H. Shrobe, R. Laddaga (Eds.), Self-Adaptive Software. Proceedings, 2000. VIII, 249 pages. 2001.

Vol. 1937: R. Dieng, O. Corby (Eds.), Knowledge Engineering and Knowledge Management. Proceedings, 2000. XIII, 457 pages. 2000. (Subseries LNAI).

Vol. 1938: S. Rao, K.I. Sletta (Eds.), Next Generation Networks. Proceedings, 2000. XI, 392 pages. 2000.

Vol. 1939: A. Evans, S. Kent, B. Selic (Eds.), «UML» – The Unified Modeling Language. Proceedings, 2000. XIV, 572 pages. 2000.

Vol. 1940: M. Valero, K. Joe, M. Kitsuregawa, H. Tanaka (Eds.), High Performance Computing. Proceedings, 2000. XV, 595 pages. 2000.

Vol. 1941: A.K. Chhabra, D. Dori (Eds.), Graphics Recognition. Proceedings, 1999. XI, 346 pages. 2000.

Vol. 1942: H. Yasuda (Ed.), Active Networks. Proceedings, 2000. XI, 424 pages. 2000.

Vol. 1943: F. Koornneef, M. van der Meulen (Eds.), Computer Safety, Reliability and Security. Proceedings, 2000. X, 432 pages. 2000.

Vol. 1944: K.R. Dittrich, G. Guerrini, I. Merlo, M. Oliva, M.E. Rodriguez (Eds.), Objects and Databases. Proceedings, 2000. X, 199 pages. 2001.

Vol. 1945: W. Grieskamp, T. Santen, B. Stoddart (Eds.), Integrated Formal Methods. Proceedings, 2000. X, 441 pages. 2000.

Vol. 1946: P. Palanque, F. Paternò (Eds.), Interactive Systems. Proceedings, 2000. X, 251 pages. 2001.

Vol. 1947: T. Sørevik, F. Manne, R. Moe, A.H. Gebremedhin (Eds.), Applied Parallel Computing. Proceedings, 2000. XII, 400 pages. 2001.

Vol. 1948: T. Tan, Y. Shi, W. Gao (Eds.), Advances in Multimodal Interfaces – ICMI 2000. Proceedings, 2000. XVI, 678 pages. 2000.

Vol. 1949: R. Connor, A. Mendelzon (Eds.), Research Issues in Structured and Semistructured Database Programming. Proceedings, 1999. XII, 325 pages. 2000.

Vol. 1950: D. van Melkebeek, Randomness and Completeness in Computational Complexity. XV, 196 pages. 2000.

Vol. 1951: F. van der Linden (Ed.), Software Architectures for Product Families. Proceedings, 2000. VIII, 255 pages. 2000.

Vol. 1952: M.C. Monard, J. Simão Sichman (Eds.), Advances in Artificial Intelligence. Proceedings, 2000. XV, 498 pages. 2000. (Subseries LNAI).

Vol. 1953: G. Borgefors, I. Nyström, G. Sanniti di Baja (Eds.), Discrete Geometry for Computer Imagery. Proceedings, 2000. XI, 544 pages. 2000.

Vol. 1954: W.A. Hunt, Jr., S.D. Johnson (Eds.), Formal Methods in Computer-Aided Design. Proceedings, 2000. XI, 539 pages. 2000.

Vol. 1955: M. Parigot, A. Voronkov (Eds.), Logic for Programming and Automated Reasoning. Proceedings, 2000. XIII, 487 pages. 2000. (Subseries LNAI).

Vol. 1956: T. Coquand, P. Dybjer, B. Nordström, J. Smith (Eds.), Types for Proofs and Programs. Proceedings, 1999. VII, 195 pages. 2000.

Vol. 1957: P. Ciancarini, M. Wooldridge (Eds.), Agent-Oriented Software Engineering. Proceedings, 2000. X, 323 pages. 2001.

Vol. 1960: A. Ambler, S.B. Calo, G. Kar (Eds.), Services Management in Intelligent Networks. Proceedings, 2000. X, 259 pages. 2000.

Vol. 1961: J. He, M. Sato (Eds.), Advances in Computing Science – ASIAN 2000. Proceedings, 2000. X, 299 pages. 2000.

Vol. 1963: V. Hlaváč, K.G. Jeffery, J. Wiedermann (Eds.), SOFSEM 2000: Theory and Practice of Informatics. Proceedings, 2000. XI, 460 pages. 2000.

Vol. 1964: J. Malenfant, S. Moisan, A. Moreira (Eds.), Object-Oriented Technology. Proceedings, 2000. XI, 309 pages. 2000.

Vol. 1965: Ç. K. Koç, C. Paar (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2000. Proceedings, 2000. XI, 355 pages. 2000.

Vol. 1966: S. Bhalla (Ed.), Databases in Networked Information Systems. Proceedings, 2000. VIII, 247 pages. 2000.

Vol. 1967: S. Arikawa, S. Morishita (Eds.), Discovery Science. Proceedings, 2000. XII, 332 pages. 2000. (Subseries LNAI).

Vol. 1968: H. Arimura, S. Jain, A. Sharma (Eds.), Algorithmic Learning Theory. Proceedings, 2000. XI, 335 pages. 2000. (Subseries LNAI).

Vol. 1969: D.T. Lee, S.-H. Teng (Eds.), Algorithms and Computation. Proceedings, 2000. XIV, 578 pages. 2000.

Vol. 1970: M. Valero, V.K. Prasanna, S. Vajapeyam (Eds.), High Performance Computing – HiPC 2000. Proceedings, 2000. XVIII, 568 pages. 2000.

Vol. 1971: R. Buyya, M. Baker (Eds.), Grid Computing – GRID 2000. Proceedings, 2000. XIV, 229 pages. 2000.

Vol. 1972: A. Omicini, R. Tolksdorf, F. Zambonelli (Eds.), Engineering Societies in the Agents World. Proceedings, 2000. IX, 143 pages. 2000. (Subseries LNAI).

Vol. 1973: J. Van den Bussche, V. Vianu (Eds.), Database Theory – ICDT 2001. Proceedings, 2001. X, 451 pages. 2001.

Vol. 1974: S. Kapoor, S. Prasad (Eds.), FST TCS 2000: Foundations of Software Technology and Theoretical Computer Science. Proceedings, 2000. XIII, 532 pages. 2000.

Vol. 1975: J. Pieprzyk, E. Okamoto, J. Seberry (Eds.), Information Security. Proceedings, 2000. X, 323 pages. 2000.

Vol. 1976: T. Okamoto (Ed.), Advances in Cryptology – ASIACRYPT 2000. Proceedings, 2000. XII, 630 pages. 2000.

Vol. 1977: B. Roy, E. Okamoto (Eds.), Progress in Cryptology – INDOCRYPT 2000. Proceedings, 2000. X, 295 pages. 2000.

Vol. 1978: B. Schneier (Ed.), Fast Software Encryption. Proceedings, 2000. VIII, 315 pages. 2001.

Vol. 1979: S. Moss, P. Davidsson (Eds.), Multi-Agent-Based Simulation. Proceedings, 2000. VIII, 267 pages. 2001. (Subseries LNAI).

Vol. 1983: K.S. Leung, L.-W. Chan, H. Meng (Eds.), Intelligent Data Engineering and Automated Learning – IDEAL 2000. Proceedings, 2000. XVI, 573 pages. 2000.

Vol. 1984: J. Marks (Ed.), Graph Drawing. Proceedings, 2001. XII, 419 pages. 2001.

Vol. 1985: J. Davidson, S.L. Min (Eds.), Languages, Compilers, and Tools for Embedded Systems. Proceedings, 2000. VIII, 221 pages. 2001.

Vol. 1987: K.-L. Tan, M.J. Franklin, J. C.-S. Lui (Eds.), Mobile Data Management. Proceedings, 2001. XIII, 289 pages. 2001.

Vol. 1988: L. Vulkov, J. Waśniewski, P. Yalamov (Eds.), Numerical Analysis and Its Applications. Proceedings, 2000. XIII, 782 pages. 2001.

Vol. 1989: M. Ajmone Marsan, A. Bianco (Eds.), Quality of Service in Multiservice IP Networks. Proceedings, 2001. XII, 440 pages. 2001.

Vol. 1990: I.V. Ramakrishnan (Ed.), Practical Aspects of Declarative Languages. Proceedings, 2001. VIII, 353 pages. 2001.

Vol. 1991: F. Dignum, C. Sierra (Eds.), Agent Mediated Electronic Commerce. VIII, 241 pages. 2001. (Subseries LNAI).

Vol. 1992: K. Kim (Ed.), Public Key Cryptography. Proceedings, 2001. XI, 423 pages. 2001.

Vol. 1993: E. Zitzler, K. Deb, L. Thiele, C.A.Coello Coello, D. Corne (Eds.), Evolutionary Multi-Criterion Optimization. Proceedings, 2001. XIII, 712 pages. 2001.

Vol. 1995: M. Sloman, J. Lobo, E.C. Lupu (Eds.), Policies for Distributed Systems and Networks. Proceedings, 2001. X, 263 pages. 2001.

Vol. 1997: D. Suciu, G. Vossen (Eds.), The World Wide Web and Databases. Proceedings, 2000. XII, 275 pages. 2001.

Vol. 1998: R. Klette, S. Peleg, G. Sommer (Eds.), Robot Vision. Proceedings, 2001. IX, 285 pages. 2001.

Vol. 1999: W. Emmerich, S. Tai (Eds.), Engineering Distributed Objects. Proceedings, 2000. VIII, 271 pages. 2001.

Vol. 2000: R. Wilhelm (Ed.), Informatics: 10 Years Back, 10 Years Ahead. IX, 369 pages. 2001.

Vol. 2003: F. Dignum, U. Cortés (Eds.), Agent Mediated Electronic Commerce III. XII, 193 pages. 2001. (Subseries LNAI).

Vol. 2004: A. Gelbukh (Ed.), Computational Linguistics and Intelligent Text Processing. Proceedings, 2001. XII, 528 pages. 2001.

Vol. 2006: R. Dunke, A. Abran (Eds.), New Approaches in Software Measurement. Proceedings, 2000. VIII, 245 pages. 2001.

Vol. 2007: J.F. Roddick, K. Hornsby (Eds.), Temporal, Spatial, and Spatio-Temporal Data Mining. Proceedings, 2000. VII, 165 pages. 2001. (Subseries LNAI).

Vol. 2009: H. Federrath (Ed.), Designing Privacy Enhancing Technologies. Proceedings, 2000. X, 231 pages. 2001.

Vol. 2010: A. Ferreira, H. Reichel (Eds.), STACS 2001. Proceedings, 2001. XV, 576 pages. 2001.

Vol. 2013: S. Singh, N. Murshed, W. Kropatsch (Eds.), Advances in Pattern Recognition – ICAPR 2001. Proceedings, 2001. XIV, 476 pages. 2001.

Vol. 2015: D. Won (Ed.), Information Security and Cryptology – ICISC 2000. Proceedings, 2000. X, 261 pages. 2001.

Vol. 2018: M. Pollefeys, L. Van Gool, A. Zisserman, A. Fitzgibbon (Eds.), 3D Structure from Images – SMILE 2000. Proceedings, 2000. X, 243 pages. 2001.

Vol. 2021: J. N. Oliveira, P. Zave (Eds.), FME 2001: Formal Methods for Increasing Software Productivity. Proceedings, 2001. XIII, 629 pages. 2001.

Vol. 2024: H. Kuchen, K. Ueda (Eds.), Functional and Logic Programming. Proceedings, 2001. X, 391 pages. 2001.

Vol. 2027: R. Wilhelm (Ed.), Compiler Construction. Proceedings, 2001. XI, 371 pages. 2001.

Vol. 2028: D. Sands (Ed.), Programming Languages and Systems. Proceedings, 2001. XIII, 433 pages. 2001.

Vol. 2029: H. Hussmann (Ed.), Fundamental Approaches to Software Engineering. Proceedings, 2001. XIII, 349 pages. 2001.

Vol. 2030: F. Honsell, M. Miculan (Eds.), Foundations of Software Science and Computation Structures. Proceedings, 2001. XII, 413 pages. 2001.

Vol. 2031: T. Margaria, W. Yi (Eds.), Tools and Algorithms for the Construction and Analysis of Systems. Proceedings, 2001. XIV, 588 pages. 2001.

Vol. 2034: M.D. Di Benedetto, A. Sangiovanni-Vincentelli (Eds.), Hybrid Systems: Computation and Control. Proceedings, 2001. XIV, 516 pages. 2001.

Table of Contents

A Query Language Based on the Ambient Logic

Luca Cardelli[1] and Giorgio Ghelli[2]

[1] Microsoft Research, 1 Guildhall Street, Cambridge, UK
[2] Università di Pisa, Dipartimento di Informatica, Corso Italia 40, Pisa, Italy

Abstract. The ambient logic is a modal logic proposed to describe the structural and computational properties of distributed and mobile computation. The structural part of the ambient logic is, essentially, a logic of labeled trees, hence it turns out to be a good foundation for query languages for semistructured data, much in the same way as first order logic is a fitting foundation for relational query languages. We define here a query language for semistructured data that is based on the ambient logic, and we outline an execution model for this language. The language turns out to be quite expressive. Its strong foundations and the equivalences that hold in the ambient logic are helpful in the definition of the language semantics and execution model.

1 Introduction

This work arises from the unexpected convergence of studies in two different fields: mobile computation and semistructured data.

Unstructured collections, or unstructured data, are collections that do not respect a predefined schema, and hence need to carry a description of their own structure. These are called *semistructured* when one can recognize in them some degree of homogeneity. This partial regularity makes semistructured collections amenable to be accessed through query languages, but not through query languages that have been designed to access fully structured databases. New languages are needed that are able to tolerate the data irregularity, and that can be used to query, at the same time, both data and structure. Semistructured collections are usually modeled in terms of labeled graphs, or labeled trees [3].

The ambient logic is a modal logic proposed to describe the structural and computational properties of distributed and mobile computation [10]. The logic comes equipped with a rich collection of logical implications and equivalences. The structural part of the ambient logic is, essentially, a logic designed to describe properties of labeled trees. It is therefore a good foundation for query languages for semistructured data, much in the same way as first order logic is a fitting foundation for relational query languages. First order logic is a logic of predicates (i.e. relations) and therefore it is particularly suitable to describe relational data. But, to describe tree-shaped data, we need a more suitable logic: a logic of trees or graphs.

This is an invited paper.

D. Sands (Ed.): ESOP 2001, LNCS 2028, pp. 1–22, 2001.

Here we define a query language for semistructured data that is based on the ambient logic, and we outline an execution model for this language. The language turns out to be quite expressive. Its strong foundations and the equivalences that hold in the ambient logic are helpful in the definition of the language semantics and execution model.

The paper is structured as follows. In this section we present a preview of the query language, and compare it with related proposals. In Section 2 we define the tree data model. In Section 3 we present the logic, upon which the query language, defined in Section 4, is defined. In Section 5 we present the evaluation model. In Section 6 we draw some conclusions.

1.1 A Preview

Consider the following bibliography, expressed in the syntax of our language TQL, which we explain in detail later. Informally, $a[F]$ represents a piece of data labeled a with contents F. The contents can be a collection of similar pieces of data, separated by "|". When the collection is empty, we can omit the brackets, so that, for example, $POPL[\,]$ can be written as $POPL$.

The bibliography below consists of a set of references all labeled *article*. Each entry contains a number of *author* fields, a *title* field, and possibly other fields.

$ARTICLES =$

$article[\ author[Cardelli] \mid author[Gordon] \mid title[Anytime\_Anywhere]$
$\quad \mid conference[POPL] \mid year[2000]$
$\quad \mid keyword[Ambient\_Calculus] \mid keyword[Logic]\ ] \mid$

$article[\ author[Cardelli] \mid title[Wide\_Area\_Computation]$
$\quad \mid booktitle[ICALP] \mid year[1999] \mid pages[403\text{-}444] \mid publisher[SV]\ ] \mid$

$article[\ author[Ghelli] \mid author[Pierce] \mid title[Bounded\_Existentials]$
$\quad \mid journal[TCS] \mid year[1998]\ ]$

Suppose we want to find all the papers in *ARTICLES* where one author is *Cardelli*; then we can write the following query:

$from \quad ARTICLES \vDash .article[X]$
$\quad\quad X \vDash .author[Cardelli]$
$select \quad paper[X]$

The query consists of a list of *matching expressions* contained between *from* and *select*, and a *reconstruction expression*, following *select*. The matching expressions bind X with every piece of data that is reachable from the root *ARTICLES* through an *article* path, and such that a path *author* goes from X to *Cardelli*; the answer is $paper[author[Cardelli] \mid author[Gordon] \mid \ldots] \mid paper[author[Cardelli] \mid title[Wide\ Area\ Computation] \mid \ldots]$, i.e. the first two articles in the databases, with the outer *article* rewritten as *paper*.

This query language is characterized by the fact that a matching expression is actually a logic expression combining matching and logical operators. For example, the following query combines path expressions and logical implication