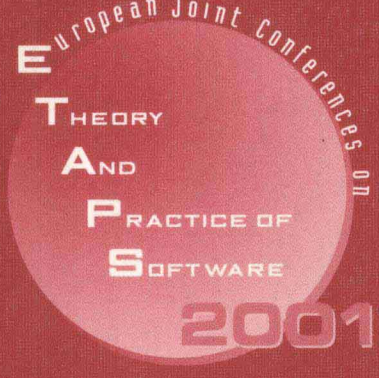
Reinhard Wilhelm (Ed.)

LNCS 2027

Compiler Construction

10th International Conference, CC 2001
Held as Part of the Joint European Conferences
on Theory and Practice of Software, ETAPS 2001
Genova, Italy, April 2001, Proceedings



Springer

Reinhard Wilhelm (Ed.)

Compiler Construction

10th International Conference, CC 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

Volume Editor

Reinhard Wilhelm
Universität des Saarlandes, Fachrichtung Informatik
Postfach 15 11 50, 66041 Saarbrücken, Germany
E-mail: wilhelm@cs.uni-sb.de

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Compiler construction : 10th international conference ; proceedings /
CC 2001, held as part of the Joint European Conferences on Theory and
Practice of Software, ETAPS 2001, Genova, Italy, April 2 - 6, 2001.
Reinhard Wilhelm (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ;
Hong Kong ; London ; Milan ; Paris ; Singapore ; Tokyo : Springer, 2001
(Lecture notes in computer science ; Vol. 2027)
ISBN 3-540-41861-X

CR Subject Classification (1998): D.3.4, D.3.1, F.4.2, D.2.6, I.2.2, F.3

ISSN 0302-9743
ISBN 3-540-41861-X 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: 10782418 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 2027

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

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

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

The International Conference on Compiler Construction (CC) is a forum for the presentation and discussion of recent developments in programming language implementation. It emphasizes practical methods and tools. CC 2001 was the tenth conference in the series.

The CC conference originated as a series of workshops started by Günter Riedewald in East Germany in 1986. In 1992 the series was relaunched by Uwe Kastens in Paderborn. In 1994 CC joined ESOP and CAAP in Edinburgh as it did 1996 in Linköping. CC federated with ESOP, FOSSACS, and TACAS to form ETAPS in 1998 and became annual. The number of submissions has shown a nice increase. The program committee received 69 submissions for CC 2001, from which 22 high-quality papers were selected for presentation. These papers are included in these proceedings. The areas of program analysis and architecture received the highest number of submissions and were rewarded with the highest number of accepted papers. Exploiting the intra-processor parallelism and improving the locality of memory referencing remain challenging problems for the compiler.

The invited speaker at CC 2001 was Ole Lehrman Madsen, whose talk was entitled *Virtual Classes and Their Implementation.* An abstract of the invited talk opens these proceedings.

The work of the CC 2001 program committee was conducted entirely by electronic means. We used the START conference management software from the University of Maryland. This proved to be very supportive for the work of the PC. Christian Probst did a remarkable job in setting it up, adding more functionality, and managing the technicalities of the submission and the reviewing process.

I am glad to acknowledge the hard work and friendly cooperation of the members of the program committee. I also wish to thank the much larger number of additional reviewers who helped us to read and evaluate the submitted papers. I appreciated very much the support and advice of the ETAPS chair, Don Sannella. Finally, I wish to thank all the authors of submitted papers for their continued interest, without which the CC conference could not thrive.

January 2001 Reinhard Wilhelm

Program Committee

Uwe Assmann (Karlsruhe)
David Bernstein (Haifa)
Evelyn Duesterwald (Cambridge, USA)
Andreas Krall (Vienna)
Rainer Leupers (Dortmund)
Mikael Petersson (Uppsala)
Martin Rinard (Cambridge, USA)
Lex Augusteijn (Eindhoven)
Stefano Crespi-Reghizzi (Milano)
Christine Eisenbeis (Rocquencourt)
Xavier Leroy (Rocquencourt)
Borivoj Melichar (Prague)
Tom Reps (Madison)
Reinhard Wilhelm (Saarbrücken) (Chair)

Referees

Giovanni Agosta
Pierre Amiranoff
Denis Barthou
Miroslav Benes
Francois Bodin
Boris Boesler
Pierre Boullier
Alessandro Campi
Zbigniew Chamski
Albert Cohen
Giuseppe Desoli
Damien Doligez
A. Ealan
Erik Eckstein
Anton Ertl
Paolo Faraboschi
Paul Feautrier
Stefan Freudenberger
Hans van Gageldonk
Thilo Gaul
Daniela Genius
David Grove
Mustafa Hagog
Dirk Heuzeroth
Paul F. Hoogendijk
Jan Hoogerbrugge
Daniel Kaestner
Felix Klock
Elliot K. Kolodner
Jaroslav Kral
Viktor Kuncak
Patrick Lam
Marc Langenbach
Sam Larsen
Sylvain Lelait
Florian Liekweg
Goetz Lindenmaier
Vassily Litvinov
Andreas Ludwig
Evelyne Lutton
Willem C. Mallon
Luc Maranget
Darko Marinov
Vincenzo Martena
Florian Martin
Nicolay Mateev
Bilha Mendelson
Andrea Ornstein
Sara Porat
Christian Probst
Ganesan Ramalingam
Martin Rinard
Erven Rohou
Radu Rugina
Petr Saloun
Pierluigi San Pietro
Bernhard Scholz
Helmut Seidl
André Seznec
Dafna Shenwald
Ron Sivan
Mark Stephenson
Henrik Theiling
Stephan Thesing
François Thomasset
Sid Ahmed Ali Touati
Joachim A. Trescher
Mon Ping Wang
Rik van de Wiel
Wim F.D. Yedema

Author Index

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

Intraprocessor Parallelism

Parsing

Memory Hierarchy

Profiling

Demos

Virtual Classes and Their Implementation

Ole Lehrmann Madsen

Computer Science Department, Aarhus University,
Åbogade 34, DK-8200 Århus N, Denmark
Ole.L.Madsen@daimi.au.dk

Abstract. One of the characteristics of BETA [4] is the unification of *abstraction* mechanisms such as class, procedure, process type, generic class, interface, etc. into one abstraction mechanism: the *pattern*. In addition to keeping the language small, the unification has given a systematic treatment of all abstraction mechanisms and leads to a number of new possibilities.

One of the interesting results of the unification is the notion of *virtual class* [7,8], which is the BETA mechanism for expressing genericity. A class may define an attribute in the form of a virtual class just as a class may define an attribute in the form of a virtual procedure. A subclass may then refine the definition of the virtual class attribute into a more specialized class. This is very much in the same way as a virtual procedure can be refined - resulting in a more specialized procedure. Virtual classes can be seen as an object-oriented version of generics. Other attempts to provide genericity for OO languages has been based on various forms of parametric polymorphism and function application rather than inheritance. Virtual classes have been used for more than 15 years in the BETA community and they have demonstrated their usefulness as a powerful abstraction mechanism. There has recently been an increasing interest in virtual classes and a number of proposals for adding virtual classes to other languages, extending virtual classes, and unifying virtual classes and parameterized classes have been made [1,2,3,13,14,15,16,17].

Another distinguishing feature of BETA is the notion of *nested class* [6]. The nested class construct originates already with Simula and is supported in a more general form in BETA. Nested classes have thus been available to the OO community for almost 4 decades, and the mechanism has found many uses in particular to structure large systems. Despite the usefulness, mainstream OO languages have not included general nesting mechanisms although C++ has a restricted form of nested classes, only working as a scoping mechanism. Recently nested classes has been added to the Java language.

From a semantic analysis point of view the combination of inheritance, and general nesting adds some complexity to the semantic analysis, since the search space for names becomes two-dimensional. With virtual classes, the analysis becomes even more complicated – for details see ref. [10].

The unification of class and procedure has also lead to *an inheritance mechanism for procedures* [5] where method-combination is based on the **inner**-construct known from Simula. In BETA, patterns are *first-class values*, which implies that procedures as well as classes are first-class values. BETA also supports the notion of *class-less* objects, which has been adapted in the form of anonymous classes in Java. Finally, it

R. Wilhelm (Ed.): CC 2001, LNCS 2027, pp. 1–2, 2001.

might be mentioned that BETA supports *coroutines* as well as *concurrent active objects*. For further details about BETA, see [6,9,11]. The Mjølner System is a program development environment for BETA and may be obtained from ref. [12].

References

1. Bruce, K., Odersky, M., Wadler, P.: A Statically Safe Alternative to Virtual Types, In: Jul, E. (ed.): 12th European Conference on Object-Oriented Programming, Brussels, June 1998. Lecture Notes in Computer Science, Vol. 1445. Springer-Verlag, Berlin Heidelberg New York, 1998.
2. Ernst, E.: Propagating Class and Method Combination, In: Guerraoui, R. (ed.): 13th European Conference on Object-Oriented Programming, Lisbon, June 1999. Lecture Notes in Computer Science, Vol. 1628. Springer-Verlag, Berlin Heidelberg New York, 1999.
3. Igarashi, A., Pierce, B.: Foundations for Virtual Types, In: Guerraoui, R. (ed.): 13th European Conference on Object-Oriented Programming, Lisbon, June 1999. Lecture Notes in Computer Science, Vol. 1628. Springer-Verlag, Berlin Heidelberg New York, 1999.
4. Kristensen, B.B., Madsen, O.L., Møller-Pedersen, B. Nygaard, K.: Abstraction Mechanisms in the BETA Programming Language. In Conference Record of the Tenth Annual ACM Symposium on Principles of Programming Languages. Austin, Texas, January 1983.
5. Kristensen, B.B., Madsen, O.L. Møller-Pedersen, B., Nygaard, K.: Classification of Actions or Inheritance also for Methods. In: Bézivin, et al. (eds.): 1st European Conference on Object-Oriented Programming, Paris, June 1987. Lecture Notes in Computer Science, Vol. 276. Springer-Verlag, Berlin Heidelberg New York, 1987.
6. Madsen, O.L., Møller-Pedersen, B., Nygaard, K.: Object-Oriented Programming in the BETA Programming Language. Addison Wesley/ACM Press, Wokingham, England, 1993. (Out of print, but a reprint can be obtained from [12]).
7. Madsen, O.L., Møller-Pedersen, B.: Virtual Classes, a Powerful Mechanism in Object-Oriented Languages. In: Proc. OOPSLA'89, New Orleans, 1989.
8. Madsen, O.L., Magnusson, B., Møller-Pedersen, B.: Strong Typing of Object-Oriented Languages Revisited. In: Proc. OOPSLA'90, Ottawa, Canada, 1990.
9. Madsen, O.L.: Open Issues in Object-Oriented Programming — A Scandinavian Perspective. Software Practice and Experience, Vol. 25, No. S4, Dec. 1995.
10. Madsen, O.L.: Semantic Analysis of Virtual Classes and Nested Classes. In: Proc. OOPSLA'99, Denver, Colorado, 1999.
11. Madsen, O.L: Towards a Unified Programming Language. In: Bertino, E. (ed.): 14th European Conference on Object-Oriented Programming. Lecture Notes in Computer Science, Vol. 1850. Springer-Verlag, Berlin Heidelberg New York, 2000.
12. The Mjølner System: http://www.mjolner.com
13. Shang, D.: Subtypes and Convertible Types, Object Currents, 1(6), June 1996.
14. Thorup. K.K.: Genericity in Java with Virtual Classes. In: Aksit, M., Matsouka, S. (eds.): 11th European Conference on Object-Oriented Programming. Lecture Notes in Computer Science, Vol. 1241. Springer Verlag, Berlin Heidelberg New York, 1997.
15. Torgersen, M.: Virtual Types are Statically Safe, 5th Workshop on Foundations of Object-Oriented Languages, January 1998.
16. Thorup, K.K., Torgersen, M.: Structured Virtual Types, informal session on types for Java, 5th Workshop on Foundations of Object-Oriented Languages, January 1998.
17. Thorup, K.K., Torgersen, M.: Unifying Genericity — Combining the benefits of Virtual types and parameterized types. In: Guerraoui, R. (ed.): 13th European Conference on Object-Oriented Programming, Lisbon, June 1999. Lecture Notes in Computer Science, Vol. 1628. Springer-Verlag, Berlin Heidelberg New York, 1999.