LNCS 4609

Erik Ernst (Ed.)

ECOOP 2007 – Object-Oriented Programming

21st European Conference
Berlin, Germany, July/August 2007
Proceedings

ecoop berlin 07

Springer

Erik Ernst (Ed.)

ECOOP 2007 – Object-Oriented Programming

21st European Conference
Berlin, Germany, July 30 - August 3, 2007
Proceedings

Springer

Volume Editor

Erik Ernst
Department of Computer Science
University of Aarhus
Denmark
E-mail: eernst@daimi.au.dk

Library of Congress Control Number: 2007931196

CR Subject Classification (1998): D.1, D.2, D.3, F.3, C.2, K.4, J.1

LNCS Sublibrary: SL 2 – Programming and Software Engineering

ISSN 0302-9743
ISBN-10 3-540-73588-7 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-73588-5 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12089611 06/3180 5 4 3 2 1 0

Lecture Notes in Computer Science 4609

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

Preface

The 21st European Conference on Object-Oriented Programming, ECOOP 2007, was held in Berlin, Germany, on July 30 to August 3, 2007. ECOOP is the most important and inspiring forum in Europe and beyond for researchers, practitioners, and students working in that smorgasbord of topics and approaches known as object orientation. This topic area was explored and challenged by excellent invited speakers—two of which were the winners of this year's Dahl-Nygaard award—in the carefully refereed and selected technical papers, on posters, via demonstrations, and in tutorials. Each of the many workshops complemented this with a very interactive and dynamic treatment of more specific topics. Finally, panels allowed for loud and lively disagreement. Yet, it is one of ECOOP's special qualities that this plethora of activities add up to a coherent and exciting whole, rather than deteriorating into chaos.

The Program Committee received 161 submissions this year. Only 135 of them were carried through the full review process, because of a number of retractions and a number of submissions of abstracts that were never followed by a full paper. However, the remaining papers were of very high quality and we accepted 25 of them for publication. Helping very good papers to be published is more useful than having an impressively low acceptance rate. The papers were selected according to four groups of criteria, whose priority depended on the paper: relevance; originality and significance; precision and correctness; and presentation and clarity. Each paper had three, four, or five reviews, depending on how controversial it was. As a new thing this year we let the authors read their reviews before the Program Committee meeting, and solicited a short response from the authors; this seemed to be helpful in several ways. The discussions at the Program Committee meeting, February 1–2 2007, were often long and agitated, but at the end we were happy with the result.

The success of ECOOP 2007 was only possible because of the dedication, inspiration, and plain hard work of many people. I would like to thank the authors for submitting so many high-quality papers. I would also like to thank the Program Committee for writing the more than 500 reviews and participating very actively in the Program Committee meeting; the Organizing Chair Stephan Herrmann for helping with numerous problems along the way; the AITO Executive board and especially Dave Thomas for their good advice on several occasions; Richard van de Stadt who was in charge of the submission Web site and its software, for his impressively quick response times, high quality of work, and generally friendly nature; and finally Karen Kjær Møller, who helped us all very much with administrative and similar tasks during the Program Committee meeting.

May 2007

Erik Ernst

Organization

ECOOP 2007 was organized by the Institute for Communication and Software Technology at the Technical University of Berlin, under the auspices of AITO (Association Internationale pour les Technologies Objets) and in cooperation with ACM SIGPLAN and SIGSOFT.

Executive Committee

Conference Chairs
  Stefan Jähnichen (Technical University of Berlin, Germany)
  Peter Fritzson (University of Linköping, Sweden)
Program Chair
  Erik Ernst (University of Aarhus, Denmark)
Organizing Chairs
  Stephan Herrmann (Technical University of Berlin, Germany)
  Uwe Aßmann (Dresden University of Technology, Germany)

Organizing Committee

Demonstration and Poster Chairs
  Sabine Glesner (Technical University of Berlin, Germany)
  Christoph Meinel (Hasso Plattner Institute, Potsdam, Germany)
Doctoral Symposium Chair
  Danny Dig (University of Illinois at Urbana-Champaign, USA)
Exhibition Chairs
  Dirk Seifert (Technical University of Berlin, Germany)
  Peter Möckel (Deutsche Telekom Laboratories, Germany)
Panel Chairs
  Bernd Mahr (Technical University of Berlin, Germany)
  Judith Bishop (University of Pretoria, South Africa)
Tutorial Chairs
  Thomas Santen (Technical University of Berlin, Germany)
  Klaus Grimm (DaimlerChrysler AG, Germany)
Workshop Chairs
  Peter Pepper (Technical University of Berlin, Germany)
  Arnd Poetzsch-Heffter (Technical University of Kaiserslautern, Germany)

Workshop Committee
- Uwe Aßmann (Dresden University of Technology, Germany)
- Lodewijk Bergmans (University of Twente, The Netherlands)
- Nick Mitchell (IBM T.J. Watson Research Center, USA)
- Mario Südholt (Ecole des Mines, Nantes-INRIA, France)
- Jan Vitek (Purdue University, USA)

Sponsoring Organizations

Gold

Silver

Bronze

Program Committee

Mehmet Aksit (University of Twente, The Netherlands)
Jonathan Aldrich (Carnegie Mellon University, USA)
Gabriela Arévalo (Universidad Nacional de La Plata, Argentina)
Uwe Aßmann (Dresden University of Technology, Germany)
Andrew P. Black (Portland State University, USA)
Gilad Bracha (Cadence Design Systems, USA)
Shigeru Chiba (Tokyo Institute of Technology, Japan)
Yvonne Coady (University of Victoria, Canada)
Krzysztof Czarnecki (University of Waterloo, Canada)
Wolfgang De Meuter (Vrije Universiteit Brussel, Belgium)
Theo D'Hondt (Vrije Universiteit Brussel, Belgium)
Stéphane Ducasse (Université de Savoie, France)

Erik Ernst (University of Aarhus, Denmark)
Yossi Gil (The Technion, Israel)
Görel Hedin (Lund University, Sweden)
Urs Hölzle (Google Inc., USA)
Eric Jul (University of Copenhagen, Denmark)
Gerti Kappel (Vienna University of Technology, Austria)
Gregor Kiczales (University of British Columbia, Canada)
Karl Lieberherr (Northeastern University, USA)
Luigi Liquori (INRIA Sophia Antipolis, France)
David H. Lorenz (University of Virginia, USA)
Sean McDirmid (École Polytechnique Fédérale de Lausanne, Switzerland)
Erik Meijer (Microsoft, USA)
Birger Møller-Pedersen (University of Oslo, Norway)
James Noble (University of Wellington, New Zealand)
Klaus Ostermann (Darmstadt University of Technology, Germany)
Awais Rashid (Lancaster University, UK)
Nathanael Schärli (Google Inc., Switzerland)
Ulrik Pagh Schultz (University of Southern Denmark, Denmark)
Olin Shivers (Northeastern University, USA)
Mario Südholt (École des Mines de Nantes, France)
Clemens Szyperski (Microsoft, USA)
Peri Tarr (IBM T.J. Watson Research Center, USA)
Frank Tip (IBM T.J. Watson Research Center, USA)
Mads Torgersen (Microsoft, USA)
Mirko Viroli (Alma Mater Studiorum - Università di Bologna, Italy)
Jan Vitek (Purdue University, USA)
Roel Wuyts (Université Libre de Bruxelles, Belgium)
Elena Zucca (University of Genova, Italy)

Referees

Ahmed Abdelmeged
Marwan Abi-Antoun
Jonathan Aldrich
Ilham Alloui
Tristan O.R. Allwood
Vander Alves
Davide Ancona
Michal Antkiewicz
Leandro Antonelli
Ivica Aracic
Federico Balaguer
Jennifer Baldwin
Andreas Bartho
Thiago Bartolomei
Nels Beckman
Alexandre Bergel
Lodewijk Bergmans
Klaas van den Berg
Robert Bialek
Kevin Bierhoff
Christoph Bockisch
Elisa Gonzalez Boix
Viviana Bono
John Boyland
Pim van den Broek
Walter Cazzola
Bryan Chadwick
Ciera Christopher
Maurizio Cimadamore
Charles Consel
Pascal Costanza
Tom Van Cutsem
Olivier Danvy
Jessie Dedecker
Birgit Demuth
Marcus Denker
Brecht Desmet
Dirk van Deun
Christophe Dony
Sophia Drossopoulou
Andrew D. Eisenberg
Torbjörn Ekman

Sonia Fagorzi
Cormac Flanagan
Manuel Fähndrich
Alessandro Garcia
Alejandra Garrido
Vaidas Gasiunas
Sofie Goderis
Ryan M. Golbeck
Aaron Greenhouse
Giovanna Guerrini
Gurcan Gulesir
Kris Gybels
Christine Hang
Wilke Havinga
Florian Heidenreich
Charlotte Herzeel
Anders Hesselund
Robert Hirschfeld
Christian Hofer
Christian Hofmann
Terry Hon
Tony Hosking
Marianne Huchard
Christian Huemer
Jendrik Johannes
Andy Kellens
Tim Klinger
Sergei Kojarski
Gerhard Kramler
Stein Krogdahl
Adrian Kuhn
Ivan Kurtev
Giovanni Lagorio
Katja Lehmann
Keren Lenz
Ondřej Lhoták
Adrian Lienhard
Chuan-kai Lin
Ralf Lämmel
Michael Maher
Donna Malayeri
Itay Maman
Chris Matthews
Weiquin Ma
Stephan Merz
Isabel Michiels
Sam Michiels
Stijn Mostinckx
Emerson Murphy-Hill
Clémentine Nebut
Oscar Nierstrasz
David Notkin
Nathaniel Nystrom
Martin Odersky
Ellen Van Paesschen
Ebraert Peter
Damien Pollet
Claudia Pons
Jean Privat
Coen De Roover
Alexandru D. Salcianu
Corina Sas
Ilie Savga
Thorsten Schäfer
Mirko Seifert
Damien Sereni
Arjun Singh
Therapon Skotiniotis
Yuiri Solodkyy
Hasan Sozer
Tom Staijen
Bjarne Stroustrup
Stan Sutton
Bedir Tekinerdogan
Stijn Timbermont
Christelle Urtado
Dale Vaillancourt
Jorge Vallejos
Sylvain Vauttier
Andrzej Wasowski
Nathan Weston
John Whaley
Manuel Wimmer
Tobias Wrigstad
Arturo Zambrano

Lecture Notes in Computer Science

For information about Vols. 1–4525

please contact your bookseller or Springer

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

Vol. 4612: I. Miguel, W. Ruml (Eds.), Abstraction, Reformulation, and Approximation. XI, 418 pages. 2007. (Sublibrary LNAI).

Vol. 4611: J. Indulska, J. Ma, L.T. Yang, T. Ungerer, J. Cao (Eds.), Ubiquitous Intelligence and Computing. XXIII, 1257 pages. 2007.

Vol. 4610: B. Xiao, L.T. Yang, J. Ma, C. Muller-Schloer, Y. Hua (Eds.), Autonomic and Trusted Computing. XVIII, 571 pages. 2007.

Vol. 4609: E. Ernst (Ed.), ECOOP 2007 — Object-Oriented Programming. XIII, 625 pages. 2007.

Vol. 4608: H.W. Schmidt, I. Crnkovic, G.T. Heineman, J.A. Stafford (Eds.), Component-Based Software Engineering. XII, 283 pages. 2007.

Vol. 4607: L. Baresi, P. Fraternali, G.-J. Houben (Eds.), Web Engineering. XVI, 576 pages. 2007.

Vol. 4606: A. Pras, M. van Sinderen (Eds.), Dependable and Adaptable Networks and Services. XIV, 149 pages. 2007.

Vol. 4605: D. Papadias, D. Zhang, G. Kollios (Eds.), Advances in Spatial and Temporal Databases. X, 479 pages. 2007.

Vol. 4604: U. Priss, S. Polovina, R. Hill (Eds.), Conceptual Structures: Knowledge Architectures for Smart Applications. XII, 514 pages. 2007. (Sublibrary LNAI).

Vol. 4603: F. Pfenning (Ed.), Automated Deduction – CADE-21. XII, 522 pages. 2007. (Sublibrary LNAI).

Vol. 4602: S. Barker, G.-J. Ahn (Eds.), Data and Applications Security XXI. X, 291 pages. 2007.

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

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

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

Vol. 4597: P. Perner (Ed.), Advances in Data Mining. XI, 353 pages. 2007. (Sublibrary LNAI).

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

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

Vol. 4594: R. Bellazzi, A. Abu-Hanna, J. Hunter (Eds.), Artificial Intelligence in Medicine. XVI, 509 pages. 2007. (Sublibrary LNAI).

Vol. 4592: Z. Kedad, N. Lammari, E. Métais, F. Meziane, Y. Rezgui (Eds.), Natural Language Processing and Information Systems. XIV, 442 pages. 2007.

Vol. 4591: J. Davies, J. Gibbons (Eds.), Integrated Formal Methods. IX, 660 pages. 2007.

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

Vol. 4589: J. Münch, P. Abrahamsson (Eds.), Product-Focused Software Process Improvement. XII, 414 pages. 2007.

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

Vol. 4587: R. Cooper, J. Kennedy (Eds.), Data Management. XIII, 259 pages. 2007.

Vol. 4586: J. Pieprzyk, H. Ghodosi, E. Dawson (Eds.), Information Security and Privacy. XIV, 476 pages. 2007.

Vol. 4585: M. Kryszkiewicz, J.F. Peters, H. Rybinski, A. Skowron (Eds.), Rough Sets and Intelligent Systems Paradigms. XIX, 836 pages. 2007. (Sublibrary LNAI).

Vol. 4584: N. Karssemeijer, B. Lelieveldt (Eds.), Information Processing in Medical Imaging. XX, 777 pages. 2007.

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

Vol. 4582: J. Lopez, P. Samarati, J.L. Ferrer (Eds.), Public Key Infrastructure. XI, 375 pages. 2007.

Vol. 4581: A. Petrenko, M. Veanes, J. Tretmans, W. Grieskamp (Eds.), Testing of Software and Communicating Systems. XII, 379 pages. 2007.

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

Vol. 4579: B. M. Hämmerli, R. Sommer (Eds.), Detection of Intrusions and Malware, and Vulnerability Assessment. X, 251 pages. 2007.

Vol. 4578: F. Masulli, S. Mitra, G. Pasi (Eds.), Applications of Fuzzy Sets Theory. XVIII, 693 pages. 2007. (Sublibrary LNAI).

Vol. 4577: N. Sebe, Y. Liu, Y.-t. Zhuang (Eds.), Multimedia Content Analysis and Mining. XIII, 513 pages. 2007.

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

Vol. 4575: T. Takagi, T. Okamoto, E. Okamoto, T. Okamoto (Eds.), Pairing-Based Cryptography – Pairing 2007. XI, 408 pages. 2007.

Vol. 4574: J. Derrick, J. Vain (Eds.), Formal Techniques for Networked and Distributed Systems – FORTE 2007. XI, 375 pages. 2007.

Vol. 4573: M. Kauers, M. Kerber, R. Miner, W. Windsteiger (Eds.), Towards Mechanized Mathematical Assistants. XIII, 407 pages. 2007. (Sublibrary LNAI).

Vol. 4572: F. Stajano, C. Meadows, S. Capkun, T. Moore (Eds.), Security and Privacy in Ad-hoc and Sensor Networks. X, 247 pages. 2007.

Vol. 4571: P. Perner (Ed.), Machine Learning and Data Mining in Pattern Recognition. XIV, 913 pages. 2007. (Sublibrary LNAI).

Vol. 4570: H.G. Okuno, M. Ali (Eds.), New Trends in Applied Artificial Intelligence. XXI, 1194 pages. 2007. (Sublibrary LNAI).

Vol. 4569: A. Butz, B. Fisher, A. Krüger, P. Olivier, S. Owada (Eds.), Smart Graphics. IX, 237 pages. 2007.

Vol. 4566: M.J. Dainoff (Ed.), Ergonomics and Health Aspects of Work with Computers. XVIII, 390 pages. 2007.

Vol. 4565: D.D. Schmorrow, L.M. Reeves (Eds.), Foundations of Augmented Cognition. XIX, 450 pages. 2007. (Sublibrary LNAI).

Vol. 4564: D. Schuler (Ed.), Online Communities and Social Computing. XVII, 520 pages. 2007.

Vol. 4563: R. Shumaker (Ed.), Virtual Reality. XXII, 762 pages. 2007.

Vol. 4562: D. Harris (Ed.), Engineering Psychology and Cognitive Ergonomics. XXIII, 879 pages. 2007. (Sublibrary LNAI).

Vol. 4561: V.G. Duffy (Ed.), Digital Human Modeling. XXIII, 1068 pages. 2007.

Vol. 4560: N. Aykin (Ed.), Usability and Internationalization, Part II. XVIII, 576 pages. 2007.

Vol. 4559: N. Aykin (Ed.), Usability and Internationalization, Part I. XVIII, 661 pages. 2007.

Vol. 4558: M.J. Smith, G. Salvendy (Eds.), Human Interface and the Management of Information, Part II. XXIII, 1162 pages. 2007.

Vol. 4557: M.J. Smith, G. Salvendy (Eds.), Human Interface and the Management of Information, Part I. XXII, 1030 pages. 2007.

Vol. 4556: C. Stephanidis (Ed.), Universal Access in Human-Computer Interaction, Part III. XXII, 1020 pages. 2007.

Vol. 4555: C. Stephanidis (Ed.), Universal Access in Human-Computer Interaction, Part II. XXII, 1066 pages. 2007.

Vol. 4554: C. Stephanidis (Ed.), Universal Acess in Human Computer Interaction, Part I. XXII, 1054 pages. 2007.

Vol. 4553: J.A. Jacko (Ed.), Human-Computer Interaction, Part IV. XXIV, 1225 pages. 2007.

Vol. 4552: J.A. Jacko (Ed.), Human-Computer Interaction, Part III. XXI, 1038 pages. 2007.

Vol. 4551: J.A. Jacko (Ed.), Human-Computer Interaction, Part II. XXIII, 1253 pages. 2007.

Vol. 4550: J.A. Jacko (Ed.), Human-Computer Interaction, Part I. XXIII, 1240 pages. 2007.

Vol. 4549: J. Aspnes, C. Scheideler, A. Arora, S. Madden (Eds.), Distributed Computing in Sensor Systems. XIII, 417 pages. 2007.

Vol. 4548: N. Olivetti (Ed.), Automated Reasoning with Analytic Tableaux and Related Methods. X, 245 pages. 2007. (Sublibrary LNAI).

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

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

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

Vol. 4544: S. Cohen-Boulakia, V. Tannen (Eds.), Data Integration in the Life Sciences. XI, 282 pages. 2007. (Sublibrary LNBI).

Vol. 4543: A.K. Bandara, M. Burgess (Eds.), Inter-Domain Management. XII, 237 pages. 2007.

Vol. 4542: P. Sawyer, B. Paech, P. Heymans (Eds.), Requirements Engineering: Foundation for Software Quality. IX, 384 pages. 2007.

Vol. 4541: T. Okadome, T. Yamazaki, M. Makhtari (Eds.), Pervasive Computing for Quality of Life Enhancement. IX, 248 pages. 2007.

Vol. 4539: N.H. Bshouty, C. Gentile (Eds.), Learning Theory. XII, 634 pages. 2007. (Sublibrary LNAI).

Vol. 4538: F. Escolano, M. Vento (Eds.), Graph-Based Representations in Pattern Recognition. XII, 416 pages. 2007.

Vol. 4537: K.C.-C. Chang, W. Wang, L. Chen, C.A. Ellis, C.-H. Hsu, A.C. Tsoi, H. Wang (Eds.), Advances in Web and Network Technologies, and Information Management. XXIII, 707 pages. 2007.

Vol. 4536: G. Concas, E. Damiani, M. Scotto, G. Succi (Eds.), Agile Processes in Software Engineering and Extreme Programming. XV, 276 pages. 2007.

Vol. 4534: I. Tomkos, F. Neri, J. Solé Pareta, X. Masip Bruin, S. Sánchez Lopez (Eds.), Optical Network Design and Modeling. XI, 460 pages. 2007.

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

Vol. 4531: J. Indulska, K. Raymond (Eds.), Distributed Applications and Interoperable Systems. XI, 337 pages. 2007.

Vol. 4530: D.H. Akehurst, R. Vogel, R.F. Paige (Eds.), Model Driven Architecture- Foundations and Applications. X, 219 pages. 2007.

Vol. 4529: P. Melin, O. Castillo, L.T. Aguilar, J. Kacprzyk, W. Pedrycz (Eds.), Foundations of Fuzzy Logic and Soft Computing. XIX, 830 pages. 2007. (Sublibrary LNAI).

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

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

Vol. 4526: M. Malek, M. Reitenspieß, A. van Moorsel (Eds.), Service Availability. X, 155 pages. 2007.

Table of Contents

Programs and Predicates

Language Design

Inheritance and Derivation

Dahl-Nygaard Prize Invited Talk

Aspects

Language About Language

Erlang – Software for a Concurrent World

Joe Armstrong

Ericsson AB
joe.armstrong@ericsson.com

Abstract. This talk is about Erlang and Concurrency Oriented Programming. We start with a short history of Erlang and of shared state and message passing concurrency. We argue that it is impossible to make fault-tolerant systems using mutable shared state concurrency models. We explain the thinking behind what has become known as "Erlang style concurrency" and show the relation to Concurrency Oriented Programming. We take a brief detour and talk about the commercial spread of Erlang, highlighting some of the more successful products and companies based on Erlang. We talk about the general problem of programming multicore computers and show how the goal of achieving factor N speedups on N-core processors with no change to the code, is being realised.

E. Ernst (Ed.): ECOOP 2007, LNCS 4609, p. 1, 2007.

Gradual Typing for Objects

Jeremy Siek[1] and Walid Taha[2]

[1] University of Colorado, Boulder, CO 80309, USA
and LogicBlox Inc., Atlanta, GA 30309, USA
jeremy.siek@colorado.edu
[2] Rice University, Houston, TX 77005, USA
taha@rice.edu

Abstract. Static and dynamic type systems have well-known strengths and weaknesses. In previous work we developed a *gradual type system* for a functional calculus named $\lambda^?_\rightarrow$. Gradual typing provides the benefits of both static and dynamic checking in a single language by allowing the programmer to control whether a portion of the program is type checked at compile-time or run-time by adding or removing type annotations on variables. Several object-oriented scripting languages are preparing to add static checking. To support that work this paper develops $\mathbf{Ob}^?_{<:}$, a gradual type system for object-based languages, extending the $\mathbf{Ob}_{<:}$ calculus of Abadi and Cardelli. Our primary contribution is to show that gradual typing and subtyping are orthogonal and can be combined in a principled fashion. We also develop a small-step semantics, provide a machine-checked proof of type safety, and improve the space efficiency of higher-order casts.

1 Introduction

Static and dynamic typing have complementary strengths, making them better for different tasks and stages of development. Static typing provides full-coverage error detection, efficient execution, and machine-checked documentation whereas dynamic typing enables rapid development and fast adaptation to changing requirements. *Gradual typing* allows a programmer to mix static and dynamic checking in a program and provides a convenient way to control which parts of a program are statically checked. The goals for gradual typing are:

- Programmers may omit type annotations on parameters and immediately run the program; run-time type checks are performed to preserve type safety.
- Programmers may add type annotations to increase static checking. When all parameters are annotated, *all* type errors are caught at compile-time.[1]
- The type system and semantics should minimize the implementation burden on language implementors.

In previous work we introduced gradual typing in the context of a functional calculus named $\lambda^?_\rightarrow$ [47]. This calculus extends the simply typed lambda calculus

[1] The language under study does not include arrays so the claim that we catch all type errors does not include the static detection of out-of-bound errors.

E. Ernst (Ed.): ECOOP 2007, LNCS 4609, pp. 2–27, 2007.

with a statically unknown (dynamic) type ? and replaces type equality with type consistency to allow for implicit coercions that add and remove ?s.

Developers of the object-oriented scripting languages Perl 6 [49] and JavaScript 4 [27] expressed interest in our work on gradual typing. In response, this paper develops the type theoretic foundation for gradual typing in object-oriented languages. Our work is based on the $\mathbf{Ob}^{<:}$ calculus of Abadi and Cardelli, a statically-typed object calculus with structural subtyping. We develop an extended calculus, named $\mathbf{Ob}^{?}_{<:}$, that adds the type ? and replaces the use of subtyping with a relation that integrates subtyping with type consistency.

The boundary between static and dynamic typing is a fertile area of research and the literature addresses many goals that are closely related to those we outline above. Section 8 describes the related work in detail.

The paper starts with a programmer's and an implementor's tour of gradual typing (Sections 2 and 3 respectively) before proceeding with the technical development of the new results in Sections 4, through 7.

Technical Contributions. This paper includes the following original contributions:

1. The primary contribution of this paper shows that type consistency and subtyping are orthogonal and can be naturally superimposed (Section 4).
2. We develop a syntax-directed type system for $\mathbf{Ob}^{?}_{<:}$ (Section 5).
3. We define a semantics for $\mathbf{Ob}^{?}_{<:}$ via a translation to the intermediate language with explicit casts $\mathbf{Ob}^{\langle\cdot\rangle}_{<:}$ for which we define a small-step operational semantics (Section 6).
4. We improve the space efficiency of the operational semantics for higher-order casts by applying casts in a lazy fashion to objects (Section 6).
5. We prove that $\mathbf{Ob}^{?}_{<:}$ is type safe (Section 7). The proof is a streamlined variant of Wright and Felleisen's syntactic approach to type soundness [5, 53]. The formalization and proof are based on a proof of type safety for $\mathbf{FOb}^{?}_{<:}$ (a superset of $\mathbf{Ob}^{?}_{<:}$ that also includes functions) we wrote in the Isar proof language [52] and checked using the Isabelle proof assistant [39]. The formalization for $\mathbf{FOb}^{?}_{<:}$ is available in a technical report [46].
6. We prove that $\mathbf{Ob}^{?}_{<:}$ is statically type safe for fully annotated programs (Section 7), that is, we show that neither cast exceptions nor type errors may occur during program execution.

2 A Programmer's View of Gradual Typing

We give a description of gradual typing from a programmer's point of view, showing examples in hypothetical variant of the ECMAScript (aka JavaScript) programming language [15] that provides gradual typing. The following `Point` class definition has no type annotations on the data member `x` or the `dx` parameter. The gradual type system therefore delays checks concerning `x` and `dx` inside the `move` method until run-time, as would a dynamically typed language.