

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

University of Dortmund, Germany

Madhu Sudan

Massachusetts Institute of Technology, MA, USA

Demetri Terzopoulos

New York University, NY, USA

Doug Tygar

University of California, Berkeley, CA, USA

Moshe Y. Vardi

Rice University, Houston, TX, USA

Gerhard Weikum

Max-Planck Institute of Computer Science, Saarbruecken, Germany

Robert Glück Michael Lowry (Eds.)

Generative Programming and Component Engineering

4th International Conference, GPCE 2005
Tallinn, Estonia, September 29 – October 1, 2005
Proceedings



Springer

Volume Editors

Robert Glück

University of Copenhagen, DIKU, Department of Computer Science

Universitetsparken 1, 2100 Copenhagen, Denmark

E-mail: glueck@acm.org

Michael Lowry

NASA Ames Research Center, M/S 269-2

Moffett Field, CA 94035-1000, USA

E-mail: lowry@email.arc.nasa.gov

Library of Congress Control Number: 2005932566

CR Subject Classification (1998): D.2, D.1, D.3, K.6

ISSN 0302-9743

ISBN-10 3-540-29138-5 Springer Berlin Heidelberg New York

ISBN-13 978-3-540-29138-1 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

springeronline.com

© Springer-Verlag Berlin Heidelberg 2005

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper SPIN: 11561347 06/3142 5 4 3 2 1 0

Preface

Generative Programming and Component Engineering (GPCE) is a leading research conference on automatic programming and component engineering. These approaches to software engineering have the potential to revolutionize software development as automation and components revolutionized manufacturing. The conference brings together researchers and practitioners interested in advancing automation for software development. It is also a premier forum for cross-fertilization between the programming language and software engineering research communities.

GPCE arose as a joint conference, merging the prior conference on Generative and Component-Based Software Engineering (GCSE) and the Workshop on Semantics, Applications, and Implementation of Program Generation (SAIG). The proceedings of the previous GPCE conferences were published in the LNCS series of Springer as volumes 2487, 2830, and 3286. In 2005 GPCE was co-located with the International Conference on Functional Programming (ICFP) and the symposium on Trends in Functional Programming (TFP), reflecting the vigorous interaction between the functional programming and generative programming research communities. GPCE and ICFP are both sponsored by the Association for Computing Machinery.

The quality and breadth of the papers submitted to GPCE 2005 was impressive. All 86 papers, including 5 papers for tool demonstrations, were rigorously reviewed by 17 highly qualified Program Committee members. The members of the Program Committee first provided in-depth individual reviews of the submitted papers, and then debated the merits of the papers through an extended electronic Program Committee meeting. After much (friendly) argument, 25 regular papers and 2 tool demonstration papers were selected for publication. The Program Committee provided extensive technical feedback to the authors of the submitted papers. The conference program was complemented with three invited talks, three extended tutorials, and three all-day workshops.

The accepted papers are grouped into eight topic areas: aspect-oriented programming, component engineering and templates, demonstrations, domain-specific languages, generative techniques, generic programming, meta-programming and transformation, and multi-stage programming. The invited talks were from leading innovators in the field: Oscar Nierstrasz on object-oriented reengineering patterns, Oege de Moor on the AspectBench compiler for AspectJ, and Bernd Fischer on certifiable program generation.

The program chairs would like to thank foremost the authors of the submitted papers: their research is the justification for this conference. Both program chairs were impressed by the expertise and diligence of the Program Committee members and their co-reviewers. Their technical dedication, as reflected in the quality of their reviews, was the foundation of the strength of these proceedings.

The general chair, Eugenio Moggi, was tireless in steering the program chairs towards a technically superb program. The publicity chair, Eelco Visser, went beyond the call of duty in raising awareness of the conference in the software engineering and programming languages research communities. Andrew Malton and Jeff Gray solicited and organized a workshop and tutorial program of interest to researchers and practitioners alike. Tarmo Uustalu graciously served as local arrangements chair, providing a hospitable atmosphere in the beautiful venue of Tallin, Estonia. The paper submissions and the reviewing process were ably supported by the Web-based EasyChair system (<http://www.easychair.org/>). The program chairs would like to extend our appreciation to Andrei Voronkov, who developed EasyChair and is the leading force behind its continued development. His personal attention to our conference greatly facilitated managing the volume of reviews and discussions amongst the Program Committee. Finally, we would like to recognize the importance of the gentle guidance of the GPCE Steering Committee. Their long-term dedication is the core that binds together this research community.

July 2005

Robert Glück
Michael Lowry

Organization

General Chair

Eugenio Moggi (Genoa University, Italy)

Program Committee Co-chairs

Robert Glück (University of Copenhagen, Denmark)

Michael Lowry (NASA Ames Research Center, USA)

Program Committee

Don Batory (University of Texas, USA)

Ira Baxter (Semantic Designs, USA)

Cristiano Calcagno (Imperial College London, UK)

Prem Devanbu (University of California at Davis, USA)

Ulrich Eisenecker (University of Leipzig, Germany)

Tom Ellman (Vassar College, USA)

Robert Filman (NASA Ames Research Center, USA)

Zhenjiang Hu (University of Tokyo, Japan)

Patricia Johann (Rutgers University, USA)

John Launchbury (Galois Connections Inc., USA)

Anne-Françoise Le Meur (University of Science and Technology Lille, France)

Hong Mei (Peking University, China)

Nicolas Rouquette (NASA Jet Propulsion Lab, USA)

William Scherlis (Carnegie Mellon University, USA)

Yannis Smaragdakis (Georgia Institute of Technology, USA)

Walid Taha (Rice University, USA)

Todd Veldhuizen (Chalmers University of Technology, Sweden)

Publicity Chair

Eelco Visser (Utrecht University, The Netherlands)

Workshop and Tutorial Chairs

Andrew Malton (University of Waterloo, Canada)

Jeff Gray (University of Alabama at Birmingham, USA)

Local Arrangements Chair

Tarmo Uustalu (Institute of Cybernetics, Estonia)

Additional Referees

Alexander Ahern
Robert L. Akers
Olivier Barais
Josh Berdine
Christie Bolton
John Tang Boyland
Dolores Diaz
Dan Dougherty
Laurence Duchien
Bernd Fischer
Aaron Greenhouse
Timothy J. Halloran
William Harrison
Jim Hook
Liwen Huang
Shan Shan Huang
Samuel Kamin
Paul Kelly
Oleg Kiselyov
Julia Lawall
Christopher League
Daan Leijen

Dongxi Liu
Jan-Willem Maessen
Michael Mehlich
Eugenio Moggi
Shin-Cheng Mu
Keisuke Nakano
Matthias Neubauer
Emir Pasalic
Renaud Pawlak
Nicolas Pessemier
Amr Sabry
Isao Sasano
Lionel Seinturier
Douglas Smith
Andreas Speck
Franklyn Turbak
Geoffrey Washburn
Tetsuo Yokoyama
Nobuko Yoshida
Hongjun Zheng
David Zook

Sponsors

GPCE 2005 was sponsored by ACM SIGPLAN, in cooperation with ACM SIGSOFT. We gratefully acknowledge the support of Utrecht University in hosting the conference Web site, and the Institute of Cybernetics, Tallinn, for handling the local arrangements and registration.

Table of Contents

Invited Talks

Object-Oriented Reengineering Patterns — An Overview <i>Oscar Nierstrasz, Stéphane Ducasse, Serge Demeyer</i>	1
abc: The AspectBench Compiler for AspectJ <i>Chris Allan, Pavel Augustinov, Aske Simon Christensen, Laurie Hendren, Sascha Kuzins, Jennifer Lhoták, Ondřej Lhoták, Oege de Moor, Damien Sereni, Ganesh Sittampalam, Julian Tibble</i> ...	10
Certifiable Program Generation <i>Ewen Denney, Bernd Fischer</i>	17

Domain-Specific Language

A Generative Programming Approach to Developing DSL Compilers <i>Charles Consel, Fabien Latory, Laurent Réveillère, Pierre Cointe</i>	29
Efficient Code Generation for a Domain Specific Language <i>Andrew Moss, Henk Muller</i>	47
On Domain-Specific Languages Reengineering <i>Christophe Alias, Denis Barthou</i>	63
Bossa Nova: Introducing Modularity into the Bossa Domain-Specific Language <i>Julia L. Lawall, Hervé Duchesne, Gilles Muller, Anne-Françoise Le Meur</i>	78

Aspect-Oriented Programming

AOP++: A Generic Aspect-Oriented Programming Framework in C++ <i>Zhen Yao, Qi-long Zheng, Guo-liang Chen</i>	94
Model Compiler Construction Based on Aspect-Oriented Mechanisms <i>Naoyasu Ubayashi, Tetsuo Tamai, Shinji Sano, Yusaku Maeno, Satoshi Murakami</i>	109
FeatureC++: On the Symbiosis of Feature-Oriented and Aspect-Oriented Programming <i>Sven Apel, Thomas Leich, Marko Rosenmüller, Gunter Saake</i>	125

Shadow Programming: Reasoning About Programs Using Lexical Join Point Information
Pengcheng Wu, Karl Lieberherr 141

Meta-programming and Transformation

Generalized Type-Based Disambiguation of Meta Programs with Concrete Object Syntax
Martin Bravenboer, Rob Vermaas, Jurgen Vinju, Eelco Visser 157

A Versatile Kernel for Multi-language AOP
Éric Tanter, Jacques Noyé 173

Semi-inversion of Guarded Equations
Torben Æ. Mogensen 189

Generative Techniques I

A Generative Programming Approach to Interactive Information Retrieval: Insights and Experiences
Saverio Perugini, Naren Ramakrishnan 205

Optimizing Marshalling by Run-Time Program Generation
Barış Aktemur, Joel Jones, Samuel Kamin, Lars Clausen 221

Applying a Generative Technique for Enhanced Genericity and Maintainability on the J2EE Platform
Yang Jun, Stan Jarzabek 237

Multi-stage Programming

Multi-stage Programming with Functors and Monads: Eliminating Abstraction Overhead from Generic Code
Jacques Carette, Oleg Kiselyov 256

Implicitly Heterogeneous Multi-stage Programming
Jason Eckhardt, Roumen Kairabachev, Emir Pašalić, Kedar Swadi, Walid Taha 275

Generative Techniques II

Source-Level Optimization of Run-Time Program Generators
Samuel Kamin, Barış Aktemur, Philip Morton 293

Statically Safe Program Generation with SafeGen
Shan Shan Huang, David Zook, Yannis Smaragdakis 309

A Type System for Reflective Program Generators <i>Dirk Draheim, Christof Lutteroth, Gerald Weber</i>	327
Sorting Out the Relationships Between Pairs of Iterators, Values, and References <i>Krister Ahlander</i>	342
Components and Templates	
Preprocessing Eden with Template Haskell <i>Steffen Priebe</i>	357
Syntactic Abstraction in Component Interfaces <i>Ryan Culpepper, Scott Owens, Matthew Flatt</i>	373
Component-Oriented Programming with Sharing: Containment is Not Ownership <i>Daniel Hirschhoff, Tom Hirschowitz, Damien Pous, Alan Schmitt, Jean-Bernard Stefani</i>	389
Generic Programming	
Language Requirements for Large-Scale Generic Libraries <i>Jeremy Siek, Andrew Lumsdaine</i>	405
Mapping Features to Models: A Template Approach Based on Superimposed Variants <i>Krzysztof Czarnecki, Michał Antkiewicz</i>	422
Demonstrations	
Developing Dynamic and Adaptable Applications with CAM/DAOP: A Virtual Office Application <i>Mónica Pinto, Daniel Jiménez, Lidia Fuentes</i>	438
Metamodeling Made Easy — MetaEdit+ <i>Risto Pohjonen</i>	442
Author Index	447