Lecture Notes in Computer Science 5969

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

Alfred Kobsa
University of California, Irvine, CA, 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
TU Dortmund University, Germany

Madhu Sudan
Microsoft Research, Cambridge, MA, USA

Demetri Terzopoulos
University of California, Los Angeles, CA, USA

Doug Tygar
University of California, Berkeley, CA, USA

Gerhard Weikum
Max-Planck Institute of Computer Science, Saarbruecken, Germany

Mark van den Brand
Dragan Gašević Jeff Gray (Eds.)

Software Language Engineering

Second International Conference, SLE 2009
Denver, CO, USA, October 5-6, 2009
Revised Selected Papers

Springer

Volume Editors

Mark van den Brand
Dept. of Mathematics and Computer Science, Software Engineering and Technology
Eindhoven University of Technology
Den Dolech 2, 5612 AZ Eindhoven, The Netherlands
E-mail: m.g.j.v.d.brand@tue.nl

Dragan Gašević
School of Computing and Information Systems
Athabasca University
1 University Drive, Athabasca, AB T9S 3A3, Canada
E-mail: dragang@athabascau.ca

Jeff Gray
Department of Computer Science
University of Alabama
P.O. Box 870290, Tuscaloosa, AL, USA
E-mail: gray@cs.ua.edu

Library of Congress Control Number: 2010922313

CR Subject Classification (1998): D.2, D.3, I.6, F.3, K.6.3

LNCS Sublibrary: SL 2 – Programming and Software Engineering

ISSN 0302-9743
ISBN-10 3-642-12106-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-12106-7 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.com

© Springer-Verlag Berlin Heidelberg 2010
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper 06/3180

Preface

We are pleased to present the proceedings of the Second International Conference on Software Language Engineering (SLE 2009). The conference was held in Denver, Colorado (USA) during October 5–6, 2009 and was co-located with the 12th IEEE/ACM International Conference on Model-Driven Engineering Languages and Systems (MODELS 2009) and the 8th ACM International Conference on Generative Programming and Component Engineering (GPCE 2009). The SLE conference series is devoted to a wide range of topics related to artificial languages in software engineering. SLE is an international research forum that brings together researchers and practitioners from both industry and academia to expand the frontiers of software language engineering.

SLE's foremost mission is to encourage and organize communication between communities that have traditionally looked at software languages from different, more specialized, and yet complementary perspectives. SLE emphasizes the fundamental notion of languages, as opposed to any realization in specific technical spaces. In this context, the term "software language" comprises all sorts of artificial languages used in software development, including general-purpose programming languages, domain-specific languages, modeling and meta-modeling languages, data models, and ontologies. Software language engineering is the application of a systematic, disciplined, quantifiable approach to the development, use, and maintenance of these languages. The SLE conference is concerned with all phases of the lifecycle of software languages; these include the design, implementation, documentation, testing, deployment, evolution, recovery, and retirement of languages. Of special interest are tools, techniques, methods, and formalisms that support these activities. In particular, tools are often based on, or automatically generated from, a formal description of the language. Hence, the treatment of language descriptions as software artifacts, akin to programs, is of particular interest—while noting the special status of language descriptions and the tailored engineering principles and methods for modularization, refactoring, refinement, composition, versioning, co-evolution, and analysis that can be applied to them.

The response to the call for papers for SLE 2009 was quite enthusiastic. We received 79 full submissions from 100 initial abstract submissions. From those 79 submissions, the Program Committee selected 23 papers: 15 full papers, 6 short papers, and 2 tool demonstration papers, resulting in an acceptance rate of 29%. To ensure the quality of the accepted papers, each submitted paper was reviewed by at least three PC members. Each paper was discussed in detail during a week-long electronic PC meeting, as facilitated by EasyChair.

The conference was quite interactive, and the discussions provided additional feedback to the authors. Accepted papers were then revised based on the reviews, in some cases a PC discussion summary, and feedback from the conference. The

final versions of all accepted papers are included in this proceedings volume. The resulting program covered diverse topics related to software language engineering. The papers cover engineering aspects in different phases of the software language development lifecycle. These include the analysis of languages in the design phase and their actual usage after deployment. The papers also represent various tools and techniques used in language implementations, including different approaches to language transformation and composition. The organization of these papers in this volume reflects the sessions in the original program of the conference.

SLE 2009 had two renowned keynote speakers: Jim Cordy (a joint keynote talk with GPCE 2009) and Jean Bézivin. They each provided informative and entertaining keynote talks. Trying to address the problems of complexity, usability, and adoption of generative and transformational techniques, Cordy's keynote suggested using generative and transformational techniques to implement domain-specific languages. Bézivin's keynote discussed the many different possibilities where model-driven research and practice can advance the capabilities for software language engineering. The proceedings begin with short papers summarizing the keynotes to provide a broad introduction to the software language engineering discipline and to identify key research challenges.

SLE 2009 would not have been possible without the significant contributions of many individuals and organizations. We are grateful to the organizers of MODELS 2009 for their close collaboration and management of many of the logistics. This allowed us to offer SLE participants the opportunity to take part in two high-quality research events in the domain of software engineering. The SLE 2009 Organizing Committee and the SLE Steering Committee provided invaluable assistance and guidance. We are especially grateful to the Software Engineering Center at the University of Minnesota for sponsoring the conference and for all the support and excellent collaboration. We must also emphasize the role of Eric Van Wyk in making this arrangement with the Software Engineering Center possible and his great help in acting as the SLE 2009 Finance Chair. We are also grateful to the PC members and the additional reviewers for their dedication in reviewing the large number of submissions. We also thank the authors for their efforts in writing and then revising their papers, and we thank Springer for publishing the papers in the proceedings. We are grateful to the developers of EasyChair for providing an open conference management system. Finally, we wish to thank all the participants at SLE 2009 for the energetic and insightful discussions that made SLE 2009 such an educational and fun event.

January 2010

Mark van den Brand
Dragan Gašević
Jeff Gray

Organization

SLE 2009 was organized by Athabasca University, Eindhoven University of Technology, and the University of Alabama. It was sponsored by the Software Engineering Center of the University of Minnesota.

General Chair

Dragan Gašević	Athabasca University, Canada

Program Committee Co-chairs

Jeff Gray	University of Alabama, USA
Mark van den Brand	Eindhoven University of Technology, The Netherlands

Organizing Committee

Alexander Serebrenik	Eindhoven University of Technology, The Netherlands (Publicity Co-chair)
Bardia Mohabbati	Simon Fraser University, Canada (Web Chair)
Marko Bošković	Athabasca University, Canada
Eric Van Wyk	University of Minnesota, USA (Finance Chair)
James Hill	Indiana University/Purdue University, USA (Publicity Co-chair)

Program Committee

Colin Atkinson	Universität Mannheim, Germany
Don Batory	University of Texas, USA
Paulo Borba	Universidade Federal de Pernambuco, Brazil
John Boyland	University of Wisconsin-Milwaukee, USA
Marco Brambilla	Politecnico di Milano, Italy
Shigeru Chiba	Tokyo Institute of Technology, Japan
Charles Consel	LaBRI / INRIA, France
Stephen Edwards	Columbia University, USA
Gregor Engels	Universität Paderborn, Germany
Robert Fuhrer	IBM Research, USA
Martin Gogolla	University of Bremen, Germany
Giancarlo Guizzardi	Federal University of Espirito Santo, Brazil
Reiko Heckel	University of Leicester, UK

Frédéric Jouault	INRIA & Ecole des Mines de Nantes, France
Nicholas Kraft	University of Alabama, USA
Thomas Kühne	Victoria University of Wellington, New Zealand
Julia Lawall	University of Copenhagen, Denmark
Timothy Lethbridge	University of Ottowa, Canada
Brian Malloy	Clemson University, USA
Kim Mens	Université catholique de Louvain, Belgium
Marjan Mernik	University of Maribor, Slovenia
Todd Millstein	University of California, Los Angeles, USA
Pierre-Etienne Moreau	Centre de recherche INRIA Nancy - Grand Est, France
Pierre-Alain Muller	University of Haute-Alsace, France
Richard Paige	University of York, UK
James Power	National University of Ireland, Ireland
Daniel Oberle	SAP Research, Germany
João Saraiva	Universidad do Minho, Portugal
Alexander Serebrenik	Eindhoven University of Technology, The Netherlands
Anthony Sloane	Macquarie University, Australia
Mary Lou Soffa	University of Virginia, USA
Steffen Staab	Universität Koblenz-Landau, Germany
Jun Suzuki	University of Massachusetts, Boston, USA
Walid Taha	Rice University, USA
Eli Tilevich	Virginia Tech, USA
Juha-Pekka Tolvanen	MetaCase, Finland
Jurgen Vinju	CWI, The Netherlands
Eelco Visser	Delft University of Technology, The Netherlands
René Witte	Concordia University, Canada

Additional Reviewers

Marcel van Amstel
Emilie Balland
Olivier Barais
Paul Brauner
Behzad Bordbar
Johan Brichau
Alfredo Cadiz
Sergio Castro
Loek Cleophas
Cristobal Costa-Soria
Duc-Hanh Dang
Adwoa Donyina
Nicolas Drivalos
João Fernandes
Frederic Fondement
Xiaocheng Ge
Danny Groenewegen
Lars Hamann
Kees Hemerik
Karsten Hoelscher
Lennart Kats
Paul Klint
Dimitros Kolovos
Mirco Kuhlmann
Nicolas Loriant
Markus Luckey
Arjan van der Meer
Muhammad Naeem
Diego Ordonez
Fernando Orejas
Nicolas Palix
Fernando Silva Perreiras
Maja Pesic
Zvezdan Protic
Alek Radjenovic
António Nestor Ribeiro
Márcio Ribeiro
Louis Rose
Christian Soltenborn

Daniel Spiewak
Tijs van der Storm
Leopoldo Teixeira
Massimo Tisi
Sander Vermolen
Nicolae Vintilla
Tobias Walter
Andreas Wübbeke
Tian Zhao

Steering Committee

Mark van den Brand	Technische Universiteit Eindhoven, The Netherlands
James Cordy	Queen's University, Canada
Jean-Marie Favre	University of Grenoble, France
Dragan Gašević	Athabasca University, Canada
Görel Hedin	Lund University, Sweden
Ralf Lämmel	Universität Koblenz-Landau, Germany
Eric Van Wyk	University of Minnesota, USA
Andreas Winter	Johannes Gutenberg-Universität Mainz, Germany

Sponsoring Institutions

Software Engineering Center

University of Minnesota
Driven to Discover℠

Table of Contents

I Keynotes

Eating Our Own Dog Food: DSLs for Generative and Transformational Engineering 1
James R. Cordy

If MDE Is the Solution, Then What Is the Problem? 2
Jean Bézivin

II Regular Papers

Session: Language and Model Evolution

Language Evolution in Practice: The History of GMF 3
Markus Herrmannsdoerfer, Daniel Ratiu, and Guido Wachsmuth

A Novel Approach to Semi-automated Evolution of DSML Model Transformation 23
Tihamer Levendovszky, Daniel Balasubramanian, Anantha Narayanan, and Gabor Karsai

Study of an API Migration for Two XML APIs 42
Thiago Tonelli Bartolomei, Krzysztof Czarnecki, Ralf Lämmel, and Tijs van der Storm

Session: Variability and Product Lines

Composing Feature Models 62
Mathieu Acher, Philippe Collet, Philippe Lahire, and Robert France

VML* – A Family of Languages for Variability Management in Software Product Lines 82
Steffen Zschaler, Pablo Sánchez, João Santos, Mauricio Alférez, Awais Rashid, Lidia Fuentes, Ana Moreira, João Araújo, and Uirá Kulesza

Multi-view Composition Language for Software Product Line Requirements 103
Mauricio Alférez, João Santos, Ana Moreira, Alessandro Garcia, Uirá Kulesza, João Araújo, and Vasco Amaral

Session: Short Papers

Yet Another Language Extension Scheme 123
Anya Helene Bagge

Model Transformation Languages Relying on Models as ADTs 133
Jerónimo Irazábal and Claudia Pons

Towards Dynamic Evolution of Domain Specific Languages 144
Paul Laird and Stephen Barrett

ScalaQL: Language-Integrated Database Queries for Scala 154
Daniel Spiewak and Tian Zhao

Integration of Data Validation and User Interface Concerns in a DSL for Web Applications 164
Danny M. Groenewegen and Eelco Visser

Ontological Metamodeling with Explicit Instantiation 174
Alfons Laarman and Ivan Kurtev

Session: Parsing, Compilation, and Demo

Verifiable Parse Table Composition for Deterministic Parsing 184
August Schwerdfeger and Eric Van Wyk

Natural and Flexible Error Recovery for Generated Parsers 204
Maartje de Jonge, Emma Nilsson-Nyman, Lennart C.L. Kats, and Eelco Visser

PIL: A Platform Independent Language for Retargetable DSLs 224
Zef Hemel and Eelco Visser

Graphical Template Language for Transformation Synthesis 244
Elina Kalnina, Audris Kalnins, Edgars Celms, and Agris Sostaks

Session: Modularity in Languages

A Role-Based Approach towards Modular Language Engineering 254
Christian Wende, Nils Thieme, and Steffen Zschaler

Language Boxes: Bending the Host Language with Modular Language Changes 274
Lukas Renggli, Marcus Denker, and Oscar Nierstrasz

Declarative Scripting in Haskell 294
Tim Bauer and Martin Erwig

Session: Metamodeling and Demo

An Automated Process for Implementing Multilevel Domain Models 314
Frédéric Mallet, François Lagarde, Charles André, Sébastien Gérard, and François Terrier

Domain-Specific Metamodelling Languages for Software Language Engineering 334
Steffen Zschaler, Dimitrios S. Kolovos, Nikolaos Drivalos, Richard F. Paige, and Awais Rashid

Generating Smart Wrapper Libraries for Arbitrary APIs 354
Uwe Jugel

Closing the Gap between Modelling and Java 374
Florian Heidenreich, Jendrik Johannes, Mirko Seifert, and Christian Wende

Author Index 385