Preface

This volume constitutes the proceedings of the workshop on Semantics, Applications, and Implementation of Program Generation (SAIG 2000) held on 20 September, 2000, in Montreal, Canada. The workshop was held as a satellite event of the ACM International Conference on Principles, Logics, and Implementations of high-level programming languages (PLI).

SAIG aims at promoting the development and the application of foundational techniques for supporting automatic program construction. As the commercial production of software systems moves further from being an art and closer to being a traditional industry, automation will necessarily play a more substantial role in the production of software, much in the same way that automation plays a crucial role in the production of other commodities, such as garments, automobiles, chemicals, and electronics.

Four prominent contributors to the area of program generation kindly agreed to deliver invited talks at SAIG 2000: Don Batory (U. Texas), Richard Kieburtz (OGI), Gilles Muller (IRISA/INRIA), and Frank Pfenning (CMU). The proceedings include abstracts of these talks.

Seven technical papers and four position papers were presented at SAIG 2000. The technical papers covered a wide spectrum of topics, including:

- Multi-stage programming languages (Calcagno and Moggi)
- Compilation of domain-specific languages and modulesystems (Elliott, Finne, and de Moore, and Helsen and Thiemann)
- Novel program transformations addressing problems specific to program generation (Makholm)
- Low-level program generation (Kamin, Callahan, and Clausen)
- Formal specification of program transformations (Fischbach and Hannon)
- Termination analysis (Song and Futamura)

The position papers also cover a broad variety of aspects of program generation, including:

- Lessons learned from previous research systems (Ramsey)
- Generation of high-performance scientific applications (Vuduc and Demmel, and Fischer, Schumann, and Pressburger)
- Type-based analysis (Berardi, Coppo, Damiani, and Giannini)

SAIG 2000 would have not been possible without the support of the PLI organizers. We would especially like to thank Amy Felty for all her effort in coordinating the PLI workshops.

July 2000

Walid Taha
Review Process

A call for papers was announced on several mailing lists and newsgroups. Twenty submissions were received, including sixteen technical submissions, and four position papers. In total, 90 reviews were written, either by Program Committee members or by the external reviewers, and each paper received at least two written reviews. The final decisions were made collectively by the Program Committee on the basis of the collected reviews. In cases where Program Committee discussions were of benefit to the authors, the discussions were summarized and included with the reviews. All submissions were of very high quality, and time constraints on the workshop were the primary reason for exclusion. In a few cases, technically outstanding papers were excluded on the basis of relevance.

Program Committee

Cliff Click (Sun)  Suresh Jagannathan (NEC)
Rowan Davies (CMU)  Tim Sheard (OGI)
Julia Lawall (DIKU)  Walid Taha (Chalmers)
Torben Mogensen (DIKU)

External Reviewers

Johan Agat  Jörgen Gustavsson  Gilles Muller
David A. Basin  Anders Hamfelt  Jörgen Fischer Nilsson
Don Batory  Beiner Häsle  Dino Oliva
Zino Bensaisa  John Hattchiff  Karol Ostrovsky
Peter Bertelsen  Boggardt Heldal  Matthew J. Saffell
Mathias Blume  Erik Hilsdale  Peter Sestoft
Magnus Carlsson  Luke Hornof  Yannis Smaragdakis
Koen Claessen  Neil Jones  Andrew Tolmach
Thierry Coquand  Sam Kamin  Philippas Tsigas
Patrick Cousot  Gregor Kiczales  Hongwei Xi
Peter Dyljar  Michael Leuschel
Marc Feeley  Manolis Marakakis
# Table of Contents

## Invited Talks

Implementing Closed Domain-Specific Languages .......................... 1  
*Richard B. Kieburz (Oregon Graduate Institute)*

Refinements and Product Line Architectures ............................. 3  
*Don Baty (University of Texas at Austin)*

Reasoning about Staged Computation .................................... 5  
*Frank Pfenning (Carnegie Mellon University)*

Specialization of Systems Programs: Lessons and Perspectives ........ 7  
*Gilles Muller (IRISA/INRIA)*

## Technical Papers

Compiling Embedded Languages ............................................. 9  
*Conal Elliott, Sigbjørn Finne (Microsoft Research), and Oege de Moor  (Oxford University)*

Lightweight and Generative Components II: Binary-Level Components ... 28  
*Sam Kamin, Miranda Callahan, and Lars Clausen (University of Illinois at Urbana-Champaign)*

Fragmental Specialization .................................................. 51  
*Simon Helsen and Peter Thiemann (University of Freiburg)*

A New Termination Approach for Specialization .......................... 72  
*Litong Song and Yoshihiko Futamura (Waseda University)*

Multi-stage Imperative Languages: A Conservative Extension Result  ... 92  
*Cristiano Calcagno and Eugenio Moggi (Università di Genova)*

Specification and Correctness of Lambda Lifting ....................... 108  
*Adam Fischbach and John Hannan (Penn State University)*

On Jones-Optimal Specialization for Strongly Typed Languages .......... 129  
*Henning Makholm (DIKU, University of Copenhagen)*

## Position Papers

Pragmatic Aspects of Reusable Program Generators ..................... 149  
*Norman Ramsey (Harvard University)*
VIII Table of Contents

Type-Based Useless-Code Elimination for Functional Programs .......... 172
  Stefano Berardi, Marrio Coppo, Ferruccio Damiani (Università di Torino),
  and Paola Giannini (Università del Piemonte Orientale)

Code Generators for Automatic Tuning of Numerical Kernels: Experiences
with FFTW ................................................................. 190
  Richard Vuduc and James W. Demmel (University of California at
  Berkeley)

Generating Data Analysis Programs from Statistical Models .......... 212
  Bernd Fischer, Johann Schumann, and Tom Pressburger (NASA Ames)

Author Index .......................................................... 231