

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

Phil Trinder Greg Michaelson
Ricardo Peña (Eds.)

Implementation of Functional Languages

15th International Workshop, IFL 2003
Edinburgh, UK, September 8-11, 2003
Revised Papers

Volume Editors

Phil Trinder
Greg Michaelson
Heriot-Watt University
School of Mathematical and Computer Sciences
Riccarton, EH14 4AS, UK
E-mail: {trinder, greg}@macs.hw.ac.uk

Ricardo Peña
Universidad Complutense de Madrid
Facultad de Informática
Departamento Sistemas Informáticos y Programación
C/ Juan del Rosal, 8, 28040 Madrid, Spain
E-mail: ricardo@sip.ucm.es

Library of Congress Control Number: 2004114139

CR Subject Classification (1998): D.3, D.1.1, F.3

ISSN 0302-9743

ISBN 3-540-23727-5 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 2004
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 11342205 06/3142 5 4 3 2 1 0

Preface

Functional programming has a long history, reaching back through early realizations in languages like LISP to foundational theories of computing, in particular λ -calculus and recursive function theory. In turn, functional programming has had wide influence in computing, both through developments within the discipline, such as formal semantics, polymorphic type checking, lazy evaluation and structural proof, and as a practical embodiment of formalized approaches, such as specification, transformation and partial application.

One of the engaging features of functional programming is precisely the crossover between theory and practice. In particular, it is regarded as essential that all aspects of functional programming are appropriately formalized, especially the specification and implementation of functional languages. Thus, specialist functional programming events like the International Workshop on the Implementation of Functional Languages (IFL) attract contributions where strong use is made of syntactic, semantic and meta-mathematical formalisms to motivate, justify and underpin very practical software systems.

IFL grew out of smaller workshops aimed at practitioners wrestling with the nuts and bolts of making concrete implementations of highly abstract languages. Functional programming has always been bedeviled by an unwarranted reputation for slow and inefficient implementations. IFL is one venue where such problems are tackled head on, always using formal techniques to justify practical implementations.

The 15th International Workshop on the Implementation of Functional Languages (IFL 2003) was held in Edinburgh, Scotland from the 8th to the 11th of September, 2003. Forty-two people attended the workshop, with participants from Australia, Germany, The Netherlands, Hungary, Ireland, Russia, Spain, Sweden and the USA, as well as from the UK.

There were 32 presentations at IFL 2003, in streams on testing, compilation and implementation, applications, language constructs and programming, types and program analysis, concurrency and parallelism, and language interfacing. Twenty-eight papers were submitted for the draft proceedings. After refereeing, 12 papers were selected for publication in these proceedings, an acceptance rate of 42%.

The Programme Committee was pleased to award the 2nd Peter Landin Prize for the best IFL paper to Pedro Vasconcelos, first author of *Inferring Costs for Recursive, Polymorphic and Higher-Order Functional Programs*¹.

The 16th International Workshop on the Implementation and Application of Functional Languages will be held in Lübeck, Germany in September 2004. For further details, please see: <http://www.isp.uni-luebeck.de/ifl104/index.htm>.

¹ Co-author Kevin Hammond honorably declined to share the prize as he was associated with its establishment.

Acknowledgments

IFL 2003 was organized by the Department of Computer Science, School of Mathematical and Computer Sciences, Heriot-Watt University.

We would like to thank June Maxwell and Christine Mackenzie for their most able workshop administration and financial arrangements. We would also like to thank Andre Rauber Du Bois for wrangling the WWW site, and Abyd Al Zain and Jann Nystrom for workshop gophering.

We are pleased to acknowledge the sponsorship of the British Computer Society Formal Aspects of Computing Special Interest Group.

May 2004

Phil Trinder, Greg Michaelson and Ricardo Peña

Programme Committee

Thomas Arts	IT-University in Gothenburg, Sweden
Clemens Grellck	University of Lübeck, Germany
Stephen Gilmore	University of Edinburgh, UK
Kevin Hammond	University of St Andrews, UK
Frank Huch	Christian Albrechts University of Kiel, Germany
Barry Jay	University of Technology, Sydney, Australia
Greg Michaelson (Chair)	Heriot-Watt University, UK
Yolanda Ortega Mallen	Universidad Complutense de Madrid, Spain
Ricardo Peña	Universidad Complutense de Madrid, Spain
Simon Peyton Jones	Microsoft Research, UK
Rinus Plasmeijer	University of Nijmegen, The Netherlands
Jocelyn Serot	Blaise Pascal University, France
Phil Trinder (Chair)	Heriot-Watt University, UK
David S. Wise	Indiana University, USA

Referees

Abdallah Al Zain	Ralf Laemmel	Clara Segura
Artem Alimarine	Hans-Wolfgang Loidl	Sjaak Smetsers
Bernd Braßel	Rita Loogen	Jonathan Sobel
Olaf Chitil	Jan Henry Nystrom	Don Syme
Koen Claessen	Enno Ohlebusch	John van Groningen
Walter Dosch	Lars Pareto	Arjen van Weelden
Andre Rauber Du Bois	Robert Pointon	Pedro Vasconcelos
David de Frutos Escrig	Fernando Rubio	
Michael Hanus	Sven-Bodo Scholz	

Sponsors



Table of Contents

Implementation of Functional Languages

I Language Constructs and Programming

Lazy Assertions	1
<i>Olaf Chitil, Dan McNeill, and Colin Runciman</i>	
Interfacing Haskell with Object-Oriented Languages	20
<i>André T.H. Pang and Manuel M.T. Chakravarty</i>	
A Functional Shell That Dynamically Combines Compiled Code	36
<i>Arjen van Weelden and Rinus Plasmeijer</i>	

II Static Analysis and Types

Polymorphic Type Reconstruction Using Type Equations	53
<i>Venkatesh Choppella</i>	
Correctness of Non-determinism Analyses in a Parallel-Functional Language	69
<i>Clara Segura and Ricardo Peña</i>	
Inferring Cost Equations for Recursive, Polymorphic and Higher-Order Functional Programs	86
<i>Pedro B. Vasconcelos and Kevin Hammond</i>	

III Parallelism

Dynamic Chunking in Eden	102
<i>Jost Berthold</i>	
With-Loop Scalarization – Merging Nested Array Operations	118
<i>Clemens Grellck, Sven-Bodo Scholz, and Kai Trojahner</i>	
Building an Interface Between Eden and Maple: A Way of Parallelizing Computer Algebra Algorithms	135
<i>Rafael Martínez and Ricardo Peña</i>	
Generic Graphical User Interfaces	152
<i>Peter Achten, Marko van Eekelen, and Rinus Plasmeijer</i>	
Polytypic Programming in Haskell	168
<i>Ulf Norell and Patrik Jansson</i>	
Author Index	185