Ralf Hinze (Ed.)

Implementation and Application of Functional Languages

24th International Symposium, IFL 2012
Oxford, UK, August 30–September 1, 2012
Revised Selected Papers

Springer
Preface

This volume contains the selected peer-reviewed revised articles that were presented at IFL 2012, the 24th International Symposium on Implementation and Application of Functional Languages. IFL 2012 was held in Oxford, UK, during 30 August–1 September, 2012. The goal of the IFL symposia is to bring together researchers actively engaged in the implementation and application of functional and function-based programming languages. IFL is a venue for researchers to present and discuss new ideas and concepts, work in progress, and publication-ripe results related to the implementation and application of functional languages and function-based programming.

The call for papers generated 37 submissions, all of which were accepted for presentation. Participants are invited to submit either a draft paper or an extended abstract describing work to be presented at the symposium. These submissions are screened by the program chair to make sure they are within the scope of IFL. It should be stressed, however, that these contributions are not peer-reviewed publications. The submissions accepted for presentation appear in the draft proceedings distributed at the symposium, which also appeared as a Department of Computer Science technical report (RR-12-06).

After the symposium, authors are given the opportunity to consider the feedback received from discussions at the symposium and are invited to submit revised full articles to the formal review process. The revised submissions are reviewed by the Program Committee considering their correctness, novelty, originality, relevance, significance, and clarity. Of the 37 papers presented at IFL 2012, the Program Committee reviewed 28 revised submissions, and 14 papers were ultimately selected for this volume. Each submission was reviewed by four members of the Program Committee. Three papers had an additional round of “shepherding” by a member of the Program Committee in order to improve the presentation and tailor it for the IFL audience.

Fritz Henglein, from the Department of Computer Science at the University of Copenhagen, was the invited speaker of IFL 2012. He delivered a truly inspiring talk about generic sorting and partitioning in linear time and fully abstractly. Thank you Fritz for your contribution to IFL 2012.

Following the IFL tradition, IFL 2012 provided participants with an opportunity to get to know each other and to talk outside the formal setting of presentations with a social event on the second day of the symposium. This year, the participants embarked on a boat trip on the river Isis, followed by a banquet at Balliol College, one of Oxford’s oldest colleges.

Carrying on a tradition started in 2003, the Program Committee selected a paper for the Peter J. Landin Award, given to the best article presented at the symposium. The recipients of the award for IFL 2012 are Kanae Tsushima and Kenichi Asai, from
Ochanomizu University, for their contribution entitled “An Embedded Type Debugger.”

Putting together IFL 2012 was truly a team effort. I am grateful to the Department of Computer Science, Elizabeth Walsh in particular, for administrative support and to St. Anne’s College for hosting the event. I would like to thank the members of the Program Committee and the external referees for their care and diligence in reviewing the submitted papers. A special thank you is due to the three shepherds, Andy Gill, Tim Sheard, and Peter Thiemann. Finally, I would like to thank Kwok-Ho Cheung, Tom Harper, Daniel James, José Pedro Magalhães, and Nicolas Wu, for their help with organizing the symposium and for distributing the call for papers. The review process and compilation of the proceedings were greatly helped by Andrei Voronkov’s EasyChair system, which I can highly recommend.

I hope that the readers will enjoy this collection of selected papers from IFL 2012. Make sure to join us at a future version of IFL!

September 2013

Ralf Hinze
Organization

Program Committee

Edwin Brady University of St. Andrews, UK
Andrew Butterfield University of Dublin, Ireland
Matthew Flatt University of Utah, USA
Andy Gill University of Kansas, USA
Stephan Herhut IntelLabs, Santa Clara, USA
Ralf Hinze University of Oxford, UK
Zhenjiang Hu National Institute of Informatics, Japan
Patrik Jansson Chalmers University of Technology, Sweden
Mauro Jaskelioff Universidad Nacional de Rosario, Argentina
Gabriele Keller University of New South Wales, Australia
Simon Marlow Microsoft Research, UK
Pablo Nogueira Technical University of Madrid, Spain
Bruno Oliveira Seoul National University, Korea
José Nuno Oliveira University of Minho, Portugal
Rinus Plasmeijer Radboud University Nijmegen, The Netherlands
Tom Schrijvers Ghent University, Belgium
Tim Sheard Portland State University, USA
Wouter Swierstra University of Utrecht, The Netherlands
Peter Thiemann University of Freiburg, Germany
Simon Thompson University of Kent, UK
Steve Zdancewic University of Pennsylvania, USA

Additional Reviewers

Kazuyuki Asada Andrew Farmer Steffen Michels
Clara Benac Earle Álvaro Fernández Díaz David Nowak
Jeroen Bransen Lars-Ake Fredlund Hugo Pacheco
Manuel Chakravarty Fritz Henglein Exequiel Rivas
Larry Diehl Pieter Koopman Neil Sculthorpe
Jonas Duregård Trevor McDonell John Van Groningen
**Local Organizing Committee**

<table>
<thead>
<tr>
<th>Name</th>
<th>Affiliation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ralf Hinze</td>
<td>University of Oxford, UK</td>
</tr>
<tr>
<td>Nicolas Wu</td>
<td>University of Oxford, UK</td>
</tr>
<tr>
<td>José Pedro Magalhães</td>
<td>University of Oxford, UK</td>
</tr>
</tbody>
</table>
Contents

A Notation for Comonads ..................................................... 1  
Dominic Orchard and Alan Mycroft

Iterating Skeletons: Structured Parallelism by Composition .......... 18  
Mischa Dieterle, Thomas Horstmeyer, Jost Berthold, and Rita Loogen

Building JavaScript Applications with Haskell ............................ 37  
Atze Dijkstra, Jurriën Stutterheim, Alessandro Vermeulen, 
and S. Doaitse Swierstra

Advances in Lazy SmallCheck ............................................. 53  
Jason S. Reich, Matthew Naylor, and Colin Runciman

OCaml-Java: From OCaml Sources to Java Bytecodes .................... 71  
Xavier Clerc

The HERMIT in the Tree: Mechanizing Program Transformations 
in the GHC Core Language ................................................ 86  
Neil Sculthorpe, Andrew Farmer, and Andy Gill

Optimisation of Generic Programs Through Inlining .................... 104  
José Pedro Magalhães

A Type- and Control-Flow Analysis for System F ........................ 122  
Matthew Fluet

Dependently-Typed Programming in Scientific Computing: 
Examples from Economic Modelling ..................................... 140  
Cezar Ionescu and Patrik Jansson

Engineering Proof by Reflection in Agda .................................. 157  
Paul van der Walt and Wouter Swierstra

Agda Meets Accelerate .................................................... 174  
Peter Thiemann and Manuel M. T. Chakravarty

An Embedded Type Debugger .............................................. 190  
Kanae Tsushima and Kenichi Asai