

Table of Contents

Introduction	1
<i>Xavier Leroy</i>	
Typed Intermediate Languages	
Compiling Java to a Typed Lambda-Calculus: A Preliminary Report	9
<i>Andrew Wright, Suresh Jagannathan, Cristian Ungureanu, Aaron Hertzmann</i>	
Stack-Based Typed Assembly Language	28
<i>Greg Morrisett, Karl Cray, Neal Glew, David Walker</i>	
How Generic Is a Generic Back End? Using MLRISC as a Back End for the TIL Compiler	53
<i>Andrew Bernard, Robert Harper, Peter Lee</i>	
Program Analyses	
A Toolkit for Constructing Type- and Constraint-Based Program Analyses (Invited Talk)	78
<i>Alexander Aiken, Manuel Fähndrich, Jeffrey S. Foster, Zhendong Su</i>	
Optimizing ML Using a Hierarchy of Monadic Types	97
<i>Andrew Tolmach</i>	
Type-Directed Continuation Allocation	116
<i>Zhong Shao, Valery Trifonov</i>	
Program Transformations and Code Generation	
Polymorphic Equality – No Tags Required	136
<i>Martin Elsman</i>	
Optimal Type Lifting	156
<i>Bratin Saha, Zhong Shao</i>	
Formalizing Resource Allocation in a Compiler	178
<i>Peter Thiemann</i>	
Memory Management	
An Approach to Improve Locality Using Sandwich Types	194
<i>Daniela Genius, Martin Trapp, Wolf Zimmermann</i>	

Garbage Collection via Dynamic Type Inference — A Formal Treatment . . 215
Haruo Hosoya, Akinori Yonezawa

Partial Evaluation and Run-Time Code Generation

Strong Normalization by Type-Directed Partial Evaluation and Run-Time
Code Generation 240
Vincent Balat, Olivier Danvy

Determination of Dynamic Method Dispatches Using Run-Time Code
Generation 253
Nobuhisa Fujinami

Distributed Computing

Type-Based Analysis of Concurrent Programs (Abstract of Invited Talk) . . 272
Naoki Kobayashi

A Type-Based Semantics for User-Defined Marshalling in Polymorphic
Languages 273
Dominic Duggan

Author Index 299