Mark Leone

School of Computer Science
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213-3891

mleone@cs.cmu.edu
office: (412) 268–2993
home: (412) 421–2502
fax: (412) 268–5576
http://www.cs.cmu.edu/~mleone

Research Interests

Programming languages, compilation, program analysis and optimization, specialization, type theory, and the application of advanced programming languages to operating systems and distributed systems.

Education

Ph.D. in Computer Science, Carnegie Mellon University	Fall 1996 (expected)
M.S. in Computer Science, Carnegie Mellon University	May 1993
B.S. with distinction (3.9 / 4.0) in Computer Science, Cornell University	May 1989

Dissertation Topic

Title: *A Principled and Practical Approach to Run-Time Code Generation*

Committee: Peter Lee (supervisor), Charles Consel, Robert Harper, Peter Steenkiste

Run-time code generation optimizes programs using values and invariants that cannot be exploited at compile time, yielding code that is often superior to statically optimal code. However, the cost of optimizing and generating code at run time can be prohibitive. In my dissertation I demonstrate how a principled use of compile-time specialization can reduce the cost of run-time code generation by an order of magnitude.

As a proof of concept, I have implemented a compiler for a subset of Standard ML that automatically produces code that generates native MIPS code at run time. This compiler has achieved significant performance improvements in real-world applications: for example, the BSD packet filter interpreter, which the BSD kernel uses for fast selection of network packets on behalf of user processes, runs nearly 40% faster on typical inputs when translated from C to ML and compiled by my system.

Publications

(http:// www.cs.cmu.edu/~ mleone/ research.html)

Peter Lee and Mark Leone. Optimizing ML with Run-Time Code Generation. In *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation*, pp. 137–148, Philadelphia, May 1996.

Mark Leone and Peter Lee. A Declarative Approach to Run-Time Code Generation. In *Workshop Record of WCSSS'96: The Inaugural Workshop on Compiler Support for System Software*, pp. 8–17, Tucson, February 1996.

Mark Leone and Peter Lee. Lightweight Run-Time Code Generation. In *Proceedings of the ACM SIGPLAN Workshop on Partial Evaluation and Semantics-Based Program Manipulation*, pages 97–106, June 1994.

Mark Leone and Peter Lee. Deferred Compilation: The Automation of Run-Time Code Generation. Technical Report CMU–CS–93–225, School of Computer Science, Carnegie Mellon University, December 1993.

Peter Lee, Mark Leone, Spiro Michaylov, and Frank Pfenning. Towards a Practical Programming Language Based on the Polymorphic Lambda Calculus. Ergo Project Report ERGO–89–085, School of Computer Science, Carnegie Mellon University, November 1989.

Research Systems

FABIUS: Designed and implemented a compiler for a subset of Standard ML that uses compile-time specialization to produce code that generates native MIPS code at run time with extremely low overhead. 1993–1996

BTA: Implemented several binding-time analyses for a higher-order functional language, based primarily on the work of Charles Consel. Spring 1993

MESS: Helped design and was the primary implementer of the MESS compiler, which explored the use of continuation passing, polymorphism, and parameterized modules in creating an intermediate representation and compiler architecture for language-independent and machine-independent compilation. (Joint work with Peter Lee et al.) 1991–1993

SML/NJ: Ported the run-time system of the Standard ML of New Jersey compiler to x86-based Unix machines. Reimplemented and tuned the x86 code generator. Fall 1992

Rosetta: Implemented an interface between the Otter resolution theorem prover and Mathematica to provide a substrate for interactive proof development and transformation. Also implemented a simple system for visualizing resolution-based proofs. (Joint work with Dana Scott et al.) Summer 1991

OVL: Implemented a front end and interpreter for an ML-like language to explore several type systems with overloading. (Joint work with Mark Lillibridge.) Fall 1990

LEAP: Added support for inductive types and type-argument synthesis to the LEAP system, an implementation of the ω-order polymorphic λ-calculus (F_ω). (Joint work with Peter Lee and Frank Pfenning.) Summer 1989

Lectures

Optimizing ML with Run-Time Code Generation
PLDI′96 Conference on Programming Language Design and Implementation — May 1996

A Declarative Approach to Run-Time Code Generation
WCSSS′96 Workshop on Compiler Support for System Software — February 1996
System Design and Implementation Seminar, Carnegie Mellon University — February 1996

Lightweight Run-Time Code Generation
PEPM′94 Workshop Partial Evaluation and Semantics-Based Program Manipulation — June 1994
Summer School on Partial Evaluation, Carnegie Mellon University — July 1994

Deferred Compilation: The Automation of Run-Time Code Generation
Principles of Programming Seminar, Carnegie Mellon University — February 1994

Overloading Resolution is Undecidable in Simple Extensions of Haskell
Ergo Seminar, Carnegie Mellon University (Joint work with Mark Lillibridge.) — September 1990

Work Experience

Engineering Co-Op, Jet Propulsion Laboratory — Fall 1987 and Summer 1988

Designed and implemented a general-purpose simulation and debugging system for validation and regression testing of the Mars Observer spacecraft's payload software. Developed hardware/software-integration test suites to verify compliance with project specifications.

Professional Activities

Referee for POPL ′93, POPL ′95, FPCA ′95, and WCSSS ′96.

Creator and maintainer of:
Resources for Programming Language Research — 1993–Present
`http://www.cs.cmu.edu/~mleone/language-research.html`
Collected Advice on Research and Writing — 1994–Present
`http://www.cs.cmu.edu/~mleone/how-to.html`

Honors

John McMullen scholar
Cornell Engineering Dean's List, eight semesters
Tau Beta Pi (lifetime member)
Sigma Xi (full member)

Additional Information

U.S. citizen

References

Peter Lee, Associate Professor
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213-3891

Email: petel@cs.cmu.edu
Phone: (412) 268–3049
Fax: (412) 268–5576

Robert Harper, Associate Professor
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213-3891

Email: rwh@cs.cmu.edu
Phone: (412) 268–3675
Fax: (412) 268–5576

Frank Pfenning, Senior Research Computer Scientist
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213-3891

Email: fp@cs.cmu.edu
Phone: (412) 268–6343 (on sabbatical until July 15)
Fax: (412) 268–5576

Olivier Danvy, Assistant Professor
Computer Science Department
Aarhus University
Ny Munkegade, Building 540
DK-8000 Aarhus C
Denmark

Email: danvy@daimi.aau.dk
Phone: (45) 89 42 33 69
Fax: (45) 89 42 32 55