

Norman Ramsey

Associate Professor
Department of Computer Science
Tufts University
Medford, Massachusetts 02155
Email: nr@cs.tufts.edu Phone: +1 617 627 4923
WWW: <http://www.cs.tufts.edu/~nr>

73 Las Casas Street
Malden, Massachusetts 02148
Phone +1 781 393 4334

Interests Programming languages and systems; functional programming

Education B.A. *summa cum laude* (physics) 1983, Princeton University.
M.S. (physics) 1986, Cornell University.
M.A. (computer science) 1990, Princeton University.
Ph.D. (computer science) 1993, Princeton University.

Honors Fannie and John Hertz Fellow, 1983–86 and 1989–91.
National Science Foundation Faculty Early Career Development Award, 1998.
Joseph R. Levenson Memorial Teaching Prize Nominee, Harvard University, 2001.
Alfred P. Sloan Research Fellow, 2001.

Employment *Computer Scientist*, Odyssey Research Associates, Ithaca, NY. 1986–88
Research Intern, DEC Systems Research Center, Palo Alto, CA. summer 1990
Research Intern, Computer Science Laboratory, Xerox PARC, Palo Alto, CA. summer 1991
Member of Technical Staff, Computer Science Research, Bellcore, Morristown, NJ. 1992–95
Visiting Assistant Professor, Department of Computer Sciences, Purdue University. 1995–96
Research Assistant Professor, Dept of Computer Science, University of Virginia. 1996–1999
Research Intern, Bell Laboratories, Murray Hill, NJ. summer 1996
Associate Professor, School of Engineering and Applied Sciences, Harvard University.
Assistant Prof. 1999–2003, Associate Prof. 2003–2008
Visiting Researcher, Microsoft Research Ltd, Cambridge, UK. summer 2007
Associate Professor, Department of Computer Science, Tufts University. since July 2008

Students Graduated João Dias, PhD, Harvard University. *Automatically Generating the Back End of a Compiler Using Declarative Machine Descriptions.* 2008
Kevin Redwine, PhD, Harvard University. *Inference Rules + Proof Strategies = Programs.* 2009

Works Edited R. Hinze and N. Ramsey, editors. *Proceedings of the 2007 ACM SIGPLAN International Conference on Functional Programming*, New York, NY, October 2007. ACM.

Refereed Conference and Workshop Publications N. Ramsey. Developing formally verified Ada programs. In *Proceedings of the 5th International Workshop on Software Specification and Design*, pages 257–265, Pittsburgh, PA, May 1989.
N. Ramsey and D. R. Hanson. A retargetable debugger. *ACM SIGPLAN '92 Conference on Programming Language Design and Implementation*, in *SIGPLAN Notices*, 27(7):22–31, July 1992.
N. Ramsey. Correctness of trap-based breakpoint implementations. In *Proceedings of the 21st ACM Symposium on the Principles of Programming Languages*, pages 15–24, Portland, OR, January 1994.
N. Ramsey and M. F. Fernández. The New Jersey Machine-Code Toolkit. In *Proceedings of the 1995 USENIX Technical Conference*, pages 289–302, January 1995.
N. Ramsey. Relocating machine instructions by currying. *Proceedings of the ACM SIGPLAN '96 Conference on Programming Language Design and Implementation*, in *SIGPLAN Notices*, 31(5):226–236, May 1996.
M. F. Fernández and N. Ramsey. Automatic checking of instruction specifications. In *Proceedings of the International Conference on Software Engineering*, pages 326–336, May 1997.
N. Ramsey and J. W. Davidson. Machine descriptions to build tools for embedded systems. In *ACM SIGPLAN Workshop on Languages, Compilers, and Tools for Embedded Systems (LCTES'98)*, LNCS 1474, pages 172–188. Springer Verlag, June 1998.

Norman Ramsey

- S. L. Peyton Jones, N. Ramsey, and F. Reig. C--: A portable assembly language that supports garbage collection. In *International Conference on Principles and Practice of Declarative Programming*, LNCS 1702, pages 1–28. Springer Verlag, September 1999. Invited paper.
- C. Cifuentes, M. Van Emmerik, and N. Ramsey. The design of a resourceable and retargetable binary translator. In *Proceedings of the Sixth Working Conference on Reverse Engineering (WCRE'99)*, pages 280–291, October 1999.
- N. Ramsey and S. L. Peyton Jones. A single intermediate language that supports multiple implementations of exceptions. *Proceedings of the ACM SIGPLAN '00 Conference on Programming Language Design and Implementation*, in *SIGPLAN Notices*, 35(5):285–298, May 2000.
- N. Ramsey and E. Csirmaz. An algebraic approach to file synchronization. In *Proceedings of the 8th European Software Engineering Conference (ESEC) and 9th ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE-9)*, pages 175–185, September 2001.
- N. Ramsey and A. Pfeffer. Stochastic lambda calculus and monads of probability distributions. *Proceedings of the 29th ACM Symposium on the Principles of Programming Languages*, in *SIGPLAN Notices*, 37(1):154–165, January 2002.
- K. Redwine and N. Ramsey. Widening integer arithmetic. In *13th International Conference on Compiler Construction (CC 2004)*, LNCS 2985, pages 232–249, April 2004.
- C. Lindig and N. Ramsey. Declarative composition of stack frames. In *13th International Conference on Compiler Construction (CC 2004)*, LNCS 2985, pages 298–312, April 2004.
- M. D. Smith, N. Ramsey, and G. Holloway. A generalized algorithm for graph-coloring register allocation. *ACM SIGPLAN '04 Conference on Programming Language Design and Implementation*, in *SIGPLAN Notices*, 39(6):277–288, June 2004.
- S. Ryu and N. Ramsey. Source-level debugging for multiple languages with modest programming effort. In *14th International Conference on Compiler Construction (CC 2005)*, pages 10–26, April 2005.
- N. Ramsey and J. Dias. An applicative control-flow graph based on Huet’s zipper. In *ACM SIGPLAN Workshop on ML*, pages 101–122, September 2005.
- N. Ramsey. ML module mania: A type-safe, separately compiled, extensible interpreter. In *ACM SIGPLAN Workshop on ML*, pages 172–202, September 2005.
- N. Ramsey, K. Fisher, and P. Govereau. An expressive language of signatures. In *Proceedings of the Tenth ACM SIGPLAN International Conference on Functional Programming (ICFP'05)*, pages 27–40, September 2005.
- R. Olinsky, C. Lindig, and N. Ramsey. Staged allocation: A compositional technique for specifying and implementing procedure calling conventions. In *Proceedings of the 33rd ACM Symposium on the Principles of Programming Languages*, pages 409–421, January 2006.
- J. Dias and N. Ramsey. Converting intermediate code to assembly code using declarative machine descriptions. In *15th International Conference on Compiler Construction (CC 2006)*, LNCS 3923, pages 217–231, March 2006.
- J. Dias and N. Ramsey. Automatically generating back ends using declarative machine descriptions. In *Proceedings of the 37th ACM Symposium on the Principles of Programming Languages*, pages 403–416, January 2010.
- N. Ramsey, J. Dias, and S. Peyton Jones. Hoopl: A modular, reusable library for dataflow analysis and transformation. In *Proceedings of the 3rd ACM SIGPLAN Symposium on Haskell (Haskell 2010)*, September 2010.
- N. Ramsey and J. Dias. Resourceable, retargetable, modular instruction selection using a machine-independent, type-based tiling of low-level intermediate code. In *Proceedings of the 38th ACM Symposium on the Principles of Programming Languages*, pages 575–586, January 2011.

Norman Ramsey

- Under Consideration** N. M. Daniels, A. Gallant, and N. Ramsey. Experience report: Haskell in computational biology. submitted to *Proceedings of the Seventeenth ACM SIGPLAN International Conference on Functional Programming (ICFP'12)*, March 2012.
- Refereed Journal Publications** N. Ramsey, E. Miron, X. Zeng, and W. Happer. Formation and breakup rates of RbXe van der Waals molecules in He and N₂ gas. *Chemical Physics Letters*, 102:340, 1983.
- N. Ramsey. Literate programming: Weaving a language-independent WEB. *Communications of the ACM*, 32(9):1051–1055, September 1989.
- N. Ramsey and C. Marceau. Literate programming on a team project. *Software—Practice & Experience*, 21(7):677–683, July 1991.
- N. Ramsey. Literate programming simplified. *IEEE Software*, 11(5):97–105, September 1994.
- N. Ramsey. A simple solver for linear equations containing nonlinear operators. *Software—Practice & Experience*, 26(4):467–487, April 1996.
- N. Ramsey and M. F. Fernández. Specifying representations of machine instructions. *ACM Transactions on Programming Languages and Systems*, 19(3):492–524, May 1997.
- N. Ramsey. Unparsing expressions with prefix and postfix operators. *Software—Practice & Experience*, 28(12):1327–1356, October 1998.
- N. Ramsey. Eliminating spurious error messages using exceptions, polymorphism, and higher-order functions. *Computer Journal*, 42(5):360–372, 1999.
- N. Ramsey and C. Cifuentes. A transformational approach to binary translation of delayed branches. *ACM Transactions on Programming Languages and Systems*, 25(2):210–224, March 2003.
- N. Ramsey. Pragmatic aspects of reusable program generators. *Journal of Functional Programming*, 13(3):601–646, May 2003. A preliminary version of this paper appeared in *Semantics, Application, and Implementation of Program Generation*, LNCS 1924, pages 149–171.
- N. Ramsey. Embedding an interpreted language using higher-order functions and types. *Journal of Functional Programming*, 21(6):585–615, November 2011. A preliminary version of this paper appeared in *Proceedings of the ACM Workshop on Interpreters, Virtual Machines, and Emulators*, June 2003.
- Technical Reports and Unrefereed Publications** N. Ramsey. A search for capacitance fluctuations in tunnel capacitors (M.S. thesis). Technical report, Materials Science Center, Cornell University, August 1986.
- N. Ramsey. A Spider user's guide. Technical Report TR-225-89, Department of Computer Science, Princeton University, August 1989.
- N. Ramsey. The Spidery WEB system of structured documentation. Technical Report TR-226-89, Department of Computer Science, Princeton University, August 1989.
- N. Ramsey. Concurrent programming in ML. Technical Report TR-262-90, Department of Computer Science, Princeton University, April 1990.
- N. Ramsey. *A Retargetable Debugger*. PhD thesis, Princeton University, Department of Computer Science, January 1993. Also Technical Report CS-TR-403-92.
- N. Ramsey and M. F. Fernández. New Jersey Machine-Code Toolkit architecture specifications. Technical Report TR-470-94, Department of Computer Science, Princeton University, October 1994. Revised December, 1996.
- N. Ramsey and M. F. Fernández. New Jersey Machine-Code Toolkit reference manual. Technical Report TR-471-94, Department of Computer Science, Princeton University, October 1994. Revised December, 1996.
- N. Ramsey and J. W. Davidson. Specifying instructions' semantics using CSDL (preliminary report). Technical Report CS-97-31, Department of Computer Science, University of Virginia, November 1997. See <http://www.cs.virginia.edu/zephyr/csdl/lrt1index.html>. Revised, May 1998.

Norman Ramsey

S. L. Peyton Jones and N. Ramsey. Machine-independent support for garbage collection, debugging, exception handling, and concurrency (draft). Technical Report CS-98-19, Department of Computer Science, University of Virginia, August 1998.

N. Ramsey and J. W. Davidson. Specifying instructions' semantics using λ -RTL (interim report). See <http://www.cs.virginia.edu/zephyr/csdl/lrtlindex.html>, December 1999.

N. Ramsey and J. K. Scott. The 1999 ICFP programming contest. *SIGPLAN Notices*, 35(3):73–83, March 2000.

K. Scott and N. Ramsey. When do match-compilation heuristics matter? Technical Report CS-2000-13, Department of Computer Science, University of Virginia, May 2000.

N. Ramsey and S. L. Peyton Jones. Featherweight concurrency in a portable assembly language. Unpublished technical report, November 2000.

C. Cifuentes, M. Van Emmerik, N. Ramsey, and B. Lewis. Experience in the design, implementation, and use of a retargetable static binary translation framework. Technical Report TR-2002-105, Sun Microsystems Laboratories, Palo Alto, CA, January 2002.

C. Cifuentes and N. Ramsey. A transformational approach to binary translation of delayed branches with applications to SPARC and PA-RISC instruction sets. Technical Report TR-2002-104, Sun Microsystems Laboratories, Palo Alto, CA, January 2002.

N. Ramsey. SIGPLAN EC activities: \LaTeX support for proceedings. *SIGPLAN Notices*, 37(4):1–3, April 2002.

N. Ramsey, J. W. Davidson, and M. F. Fernández. Design principles for machine-description languages. Unpublished draft available at <http://www.cs.tufts.edu/~nr/pubs/desprin-abstract.html>, 2001.

N. Ramsey and C. Lindig. Custom calling conventions in a portable assembly language. Unpublished paper available at <http://www.cs.tufts.edu/~nr/pubs/custom-abstract.html>, November 2002.

S. Ryu and N. Ramsey. The 1db interface. Technical Report TR-23-04, Division of Engineering and Applied Sciences, Harvard University, February 2005.

N. Ramsey, S. Peyton Jones, and C. Lindig. The C-- language specification Version 2.0 (CVS revision 1.128). See <http://www.cminusminus.org/code.html#spec>, February 2005.

N. Ramsey. Teach technical writing in two hours per week, October 2006. A handbook for teaching writing to students in science or engineering. Accompanied by a student's edition.

J. Dias and N. Ramsey. Automatically generating back ends for a portable assembly language using declarative machine descriptions. Technical report, Harvard University, January 2008.

N. Ramsey, J. Dias, and S. Peyton Jones. Hoopl: Dataflow optimization made simple. July 2009.

N. Ramsey. Teaching technical writing using the engineering method, March 2011. A handbook for teaching writing to students in science or engineering. Accompanied by a student's edition.

Program Committees and Review Committees

Program committee, ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI). 1998 and 2002

Review panel, National Science Foundation. 1999, 2001, 2002, and 2009

Program committee, ACM Workshop on Partial Evaluation and Program Manipulation (PEPM). 2000

Program committee, Workshop on Binary Translation. 2001

Program committee, Compiler Construction. 2002

Program committee, International Conference on Functional Programming (ICFP). 2002 and 2005

Program committee, ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI). 2007

Program chair, International Conference on Functional Programming (ICFP). 2007

Program committee, ACM SIGPLAN Workshop on ML. 2009

Program committee, ACM Symposium on the Principles of Programming Languages (POPL). 2012

Norman Ramsey

Other Professional Service	Conference committee (programming contest), International Conference on Functional Programming (ICFP). 1999
	Member at large, SIGPLAN Executive Committee. 2001–03
	Selection committee, SIGPLAN Programming Languages Achievement Award. 2002 and 2008
	Editorial board, Journal of Functional Programming. since 2004
	Steering committee, International Conference on Functional Programming (ICFP). 2006–2009
	Editor, Journal of Functional Programming, special issue of invited papers from ICFP 2007 (jointly with Mark P. Jones). 2008–2009
Sponsored Research	<i>Object-code Instrumentation for Test-Coverage Analysis</i> , Software Engineering Research Center, \$51,341. 1996
	<i>Migration of Legacy Software by Binary Translation</i> , Australian Research Council, \$126,000 (with Cristina Cifuentes). 1997–99
	<i>Reusable Specifications for Retargetability</i> , National Science Foundation, Faculty Early Career Development (CAREER), \$200,000. 1998–2002
	<i>C-- Development</i> , Microsoft Research, \$100,000. 1999–2000
	<i>C-- Development (renewal)</i> , Microsoft Research, \$100,000. 2001–2002
	<i>Active Debugging Information for Multi-Platform, Multi-Language Debugging</i> , National Science Foundation, Next Generation Software, \$499,082. 1999–2002
	<i>Using Declarative Machine Descriptions in a Retargetable Optimizing Compiler</i> , National Science Foundation, Compilers, \$350,000. 2003–2007
	<i>A Reusable, Extensible, Optimizing Back End</i> , National Science Foundation, Information Technology Research, \$598,039. 2003–2009
	<i>Multicore Support in Programming-Language Infrastructure</i> , Intel, \$50,000. 2006–2007
	<i>Multicore Support in Programming-Language Infrastructure (renewal)</i> , Intel, \$50,000. 2009–2010
Other Grants	<i>Interpreters for Teaching Programming Languages</i> , Dean for Undergraduate Education, Harvard University, \$8,550. 2000
	<i>Inter-Document Cross-Reference for Texts of Large Programs</i> , Clark fund, Harvard University, \$3,997. 2001–2002
	<i>Teach Technical Writing in Two Hours Per Week</i> , Gordon Gray Faculty Grant for Writing Pedagogy, Harvard University, \$5,000. 2003–2004
	<i>Functional Foundations for Configuration Management and Source-Code Control</i> , Grant-in-aid, Tufts Faculty Research Awards Committee, \$1,432.65. 2011
Selected Invited Presentations	“Literate Programming Should be a Model for Software Engineering and Programming Languages.” ICSE-17 Workshop on Software Engineering and Programming Languages April 24–25, 1995
	“A Single Intermediate Language That Supports Multiple Implementations of Exceptions” Dagstuhl Seminar 0045.1 (Effective Implementation of Object-Oriented Languages) November 5–8, 2000
	“Engineering Calling Conventions in the Quick C-- Compiler Infrastructure” New England Programming Languages Symposium, Brandeis University, Waltham, Mass. October 21, 2003
	“Embedding an Interpreted Language Using Higher-Order Functions and Types” MIT Workshop on Lightweight Languages (LL3), Cambridge, Mass. November 8, 2003
	“The C-- Compiler Infrastructure” (with Simon L. Peyton Jones) Tutorial, ACM SIGPLAN Conference on Programming Language Design and Implementation June 8, 2004
	Tutorial, Ninth ACM SIGPLAN International Conference on Functional Programming September 19, 2004

Norman Ramsey

“Building the World from First Principles: Declarative Machine Descriptions and Compiler Construction”

Plenary invited talk, Seventh International Symposium on Practical Aspects of Declarative Languages
January 12-14, 2005

“Seeking the Best Simple Interpreter”

IFIP Working Group 2.8 (Functional Programming), Dedham, Mass. *July 17, 2006*

“Generic Typed Intermediate Languages for the Masses”

IFIP Working Group 2.8 (Functional Programming), Nesjavellir, Iceland. *July 16, 2007*

“Dataflow Optimization for the Semantically Inclined”

IFIP Working Group 2.8 (Functional Programming), Park City, Utah. *June 16, 2008*

“Can Functional Programmers Make ~~make~~ Make Sense?”

IFIP Working Group 2.8 (Functional Programming), Marble Falls, Texas.
March 10, 2011

Courses Taught

Programming languages (graduate and undergraduate), machine-level programming (undergraduate), systems programming (undergraduate), concurrent programming (graduate), advanced functional programming (graduate), compilers (graduate), dataflow analysis and optimization (graduate), advanced systems programming (graduate), operating systems and networks (undergraduate), technical writing (mixed undergraduate and graduate).

Citizenship

United States.