

Brian Richard Murphy
brm@cs.stanford.edu

Mailing Address

809 Shary Ave
Mountain View, CA 94041
mobile: +1-650-714-8003

Work Address

Intel Labs
2200 Mission College Blvd. SC12-303
Santa Clara, CA 95054
fax: +1-650-745-0982

Education

STANFORD UNIVERSITY Stanford, CA
Received Ph.D. in Computer Science in January, 2002, under Prof. Monica Lam, with dissertation *Frameworks for Precise Program Analysis*. Research includes program analysis for parallelizing compilers, advanced program analysis techniques.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY Cambridge, MA
Received B.S. in Computer Science and Engineering and M.S. in Electrical Engineering and Computer Science in September, 1990.

Experience

Intel Corporation Santa Clara, CA
Just-in-time Researcher (Feb 2000-): Developed high-level analyses and optimizations for a multi-language, multi-target just-in-time compiler. Personally originated at least one novel and patentable technique. Concurrently assisted several short-term special-purpose projects, e.g.: to improve existing JIT performance, involving both analysis and low-level instrumentation and performance optimization.

Eliren Intl., Inc. Menlo Park, CA and Beijing, China
Chief Technical Officer (2000-2001), Board Director (2001): Supervised technical development and operations for internet activities (content/community web site, online games) of this multimedia startup company, which produces online, print, and TV content for contemporary Chinese women. Supervised negotiations with providers. Directed technical staff and programmers, as well as personally handling much computer system, web site, and database configuration, administration, integration, and tuning. In the early days, experimented with candidate systems for web site architecture and content update. Advised CEO and founder about technical issues.

Stanford University Stanford, CA
Research Assistant (1990-2001): Co-developed, implemented, and maintained interprocedural program analysis system, and a number of program analyses, to facilitate automatic parallelization of Fortran programs. Implemented an analysis which combined predicate information with scalar privatization analysis, and generalized this to a theory which allowed implementation of further analyses to allow conditional parallelization of some codes. Generalized the technique of analysis using partial transfer functions (from a single prior instance), and developed theory to allow its application to additional analyses.

Collaborated on programming, documentation, and release engineering for the SUIF National Compiler Infrastructure project, and its predecessor, the SUIF 1 compiler system. Administered various computer systems and software for research group. Extensive C++ design and coding, Perl scripting.

Teaching Assistant (SPRING 1995, 1996, WINTER 1993): Topics in Compilers, Advanced Compilation Techniques.

I.B.M. Almaden Research Center San Jose, CA
Functional Languages Group (1989-1990, 1991): Designed and implemented a type

inference system for the FL language based upon abstract interpretation. Developed FL-to-C compiler prototype implemented in Common Lisp.

(SUMMER 1987, 1988): Contributed to the design of the computer language FL. Completed and maintained a compiler for FL in Lisp through many language changes. Explored issues related to the optimization of FL programs by algebraic rewriting.

M.I.T. Project Athena – SYSTEMS PROGRAMMER (1986–1987) Cambridge, MA

Datamap – PROGRAMMING CONSULTANT (1984–1985) Upper Darby, PA

Skills

Much experience with compilers, program analysis, and programming language design and implementation. Extensive programming experience in C++, Perl, C, Clu, Common Lisp, Fortran, Pascal, Scheme, TCL, various assembler languages. Exposure to Java, ML, PHP, PL/SQL, etc. Systems programming/administration of Unix (BSD 4.3, Linux, Solaris, Digital Unix, SGI Irix). Web site admin/programming (Apache, CGI/mod_perl, MySQL, Oracle, AOLserver). Minimal spoken Chinese. Some management experience with a small team (6-10 people).

Honors

AT&T Bell Labs Doctoral Scholar (1991–1995). Tau Beta Pi and Eta Kappa Nu honor societies (1989). High school valedictorian (out of 599). National Merit Scholar, Philadelphia Science Council Award (1985).

Publications

B. R. Murphy, *Frameworks for Precise Program Analysis*. Doctoral Dissertation, Stanford University, 2001.

M. W. Hall, S. P. Amarasinghe, B. R. Murphy, S. Liao, and M. S. Lam, Interprocedural Parallelization Analysis in SUIF. To appear in *ACM Transactions of Programming Language and Systems*.

B. R. Murphy and M. S. Lam, Program analysis using partial transfer functions. In *Workshop on Partial Evaluation and Program Manipulation*, Jan, 2000.

J. Anderson, M. Hall, S. Amarasinghe, B. Murphy, S. Liao, E. Bugnion, and M. Lam, Achieving High Performance on Digital AlphaServers with the SUIF Compiler. *Digital Technical Journal*, Vol. 10 No. 1, 1998, pages 71-80.

S. Moon, M. W. Hall, and B. R. Murphy, Predicated Array Data-Flow Analysis for Runtime Parallelization. In *International Conference on Supercomputing*, July, 1998.

M.W. Hall, J.M. Anderson, S.P. Amarasinghe, B.R. Murphy, S. Liao, E. Bugnion, and M. Lam, Maximizing Multiprocessor Performance with the SUIF Compiler. *IEEE Computer*, December, 1996.

S. P. Amarasinghe, J. M. Anderson, C. S. Wilson, S.-W. Liao, B. R. Murphy, R. S. French, M. W. Hall, and M. S. Lam, Multiprocessors from a Software Perspective. *IEEE Micro*, 16(3), June 1996.

M. W. Hall, S. P. Amarasinghe, B. R. Murphy, S. Liao, and M. S. Lam, Detecting Coarse-Grain Parallelism Using an Interprocedural Parallelizing Compiler. In *Proceedings of Supercomputing '95*, December, 1995.

M. W. Hall, B. R. Murphy, S. P. Amarasinghe, S. Liao, and M. S. Lam, Interprocedural Analysis for Parallelization. In *8th International Workshop on Languages and Compilers for Parallel Computing (LCPC95)*. Springer-Verlag, August, 1995.

M. W. Hall, B. R. Murphy, and S. Amarasinghe, Interprocedural Parallelization Analysis: A Case Study. In *Proceedings of the Seventh SIAM Conference on Parallel Processing for Scientific Computing*, February, 1995.

A. Aiken and B. R. Murphy, Implementing Regular Tree Expressions. In *5th ACM Symposium on Functional Programming Languages and Computer Architectures*, August, 1991.

A. Aiken and B. R. Murphy, Static Type Inference in a Dynamically Typed Language. In *18th ACM Symposium on Principles of Programming Languages*, Orlando, FL, 1991.

B. R. Murphy, *A Type Inference System for FL*. Master's thesis, MIT, 1990.