John Whaley

Gates Building, Room 406   Email: jwhaley@alum.mit.edu
353 Serra Mall   URL: http://www.stanford.edu/~jwhaley
Stanford, CA 94305
Tel: (650) 725-3720   Fax: (650) 725-6949

Research Interests

Program analysis, compilers, programming languages, software engineering, pointer analysis, virtual machines, binary decision diagrams, bug finding, program transformations.

Education

Ph.D. Computer Science, June 2005. (expected)   Stanford University, Stanford, CA
   Advisor: Monica S. Lam
   Thesis: Program Analysis using Binary Decision Diagrams

M.Eng. Electrical Engineering & Computer Science, 1999. Massachusetts Institute of Technology,
Advisor: Martin Rinard   Cambridge, MA
   Thesis: Dynamic Optimization through the Use of Automatic Runtime Specialization

B.S. Computer Science and Engineering, 1999.   Massachusetts Institute of Technology,
   Minor: East Asian Studies   Concentration: Japanese   Cambridge, MA
   Cumulative GPA: **4.9 / 5.0**   GPA in major: **5.0 / 5.0**

Research Experience

Research Assistant for Prof. Monica Lam, Stanford University   *September 2000 to present*
I performed research on a wide variety of topics, many of which appeared in top conferences:
- Developed bddbddb (BDD-Based Deductive DataBase), an implementation of Datalog that represents relations as binary decision diagrams. I used bddbddb to implement context-sensitive pointer analysis, among other analyses [PLDI 2004 Best Paper].
- Investigated using machine learning techniques to find effective BDD variable orders for BDD-based program analysis.
- Developed and evaluated a set of heuristics for method-level speculative parallelization.
- Designed and wrote Joeq, an advanced compiler and virtual machine infrastructure that is used by numerous researchers throughout the world and as the basis for the compilers course at Stanford [IVME 2003, SCP 2005].
- Implemented a fast inclusion-based context-insensitive pointer analysis for Java [SAS2002].
- Researched techniques for automatically extracting component interfaces using static and dynamic analysis [ISSTA 2002 Distinguished Paper].
- Developed a technique for system checkpointing using reflection and program analysis [Reflection 2001].

Visiting Researcher, IBM Tokyo Research Laboratory   *August 1999 to August 2000*
   I worked on the IBM product JIT compiler, interacting with groups in Toronto, New York, Hursley, and Haifa. I implemented a context-sensitive sampling profiler and developed the technique of partial method compilation. [Java Grande 2000, OOPSLA 2001 Best Paper]

Internship, IBM Watson Research Laboratory   *January 1998 to August 1998*
   I designed and implemented the optimizing compiler for the Jalapeño Java virtual machine (now called Jikes RVM). I also participated in the design of many other components of the system. My work on Jalapeño culminated into my Masters thesis at MIT. [Java Grande 1999, IBMSJ 2000]

Real-Time Manipulation of Objects in a Ray Traced Scene   *Fall 1997*
    I conceived, designed, and implemented an algorithm for real time ray tracing. It runs full screen and real time on a standard Pentium. Submitted to SIGGRAPH 99.

Internship, IBM Watson Research Laboratory   *Summer 1997*
    I worked on a clean-room Java virtual machine implementation written in C++. I rewrote the JIT compiler and implemented a real-time profiler, which was featured in an article in Dr. Dobbs.

Robotic Dog with Speech Control   *Spring 1997*
    I designed and constructed a robotic dog with an adaptive speech recognition system using only low-cost discrete components. This dog won the Bell Northern Outstanding Research Prize.

Internship, IBM Watson Research Laboratory   *Summer 1996*
    I worked in the Mobile Solutions group designing handheld devices. I implemented an extension to Java to support pen input and playback.

Research Assistant, MIT AI Laboratory   *January 1996*
    I worked on a robot that used visual-based intelligence with a natural language interface.

Teaching Experience

Teaching Assistant, Stanford, Winter 2004. CS240: Advanced Topics in Operating Systems.
Teaching Assistant, Stanford, Winter 2002. CS243: Advanced Compiling Techniques.
Teaching Assistant, Stanford, Spring 2001. CS343: Advanced Topics in Compilers.
Teaching Assistant, MIT, Fall 1998. 6.035: Computer Language Engineering.
Grader, MIT, 1997-1998. 1.00: Introduction to Computers and Engineering Problem Solving.
Laboratory Assistant, MIT, 1995-1997. 6.001: Structure and Interpretation of Computer Programs.

Refereed Publications

Whaley. Joeq: A Virtual Machine and Compiler Infrastructure. Science of Computer Programming Journal (SCP). To appear in 2005. This is an extended version of my IVME paper.

Whaley, Lam. Cloning-Based Context-Sensitive Pointer Alias Analyses Using Binary Decision Diagrams. Programming Languages Design and Implementation (PLDI 2004), Washington, DC. **PLDI Best Paper Award.**

Whaley. Joeq: A Virtual Machine and Compiler Infrastructure. Workshop on Interpreters, Virtual Machines, and Emulators (IVME 2003), San Diego, CA.

Whaley, Lam. An Efficient Inclusion-Based Points-To Analysis for Strictly-Typed Languages. Static Analysis Symposium (SAS 2002), Madrid, Spain.

Whaley, Martin, Lam. Automatic Extraction of Object-Oriented Component Interfaces. International Symposium on Software Testing and Analysis (ISSTA 2002), Rome, Italy. **ACM SIGSOFT Distinguished Paper Award.**

Whaley. Partial Method Compilation using Dynamic Profile Information. Object-Oriented Programming Systems, Languages, and Applications (OOPSLA 2001), Tampa, FL. **OOPSLA Best Paper Award.**

Whaley. System Checkpointing Using Reflection and Program Analysis. Reflection, 3rd International Conference on Metalevel Architectures and Separation of Crosscutting Concerns, 2001, Kyoto, Japan.

Whaley. A Portable Sampling-Based Profiler for Java Virtual Machines. In ACM 2000 Java Grande Conference, San Francisco, CA.

Alpern, Attanasio, Barton, Burke, Cheng, Choi, Cocchi, Fink, Grove, Hind, Hummel, Lieber, Litvinov, Ngo, Mergen, Sarkar, Serrano, Shepherd, Smith, Sreedhar, Srinivasan, Whaley. The Jalapeno Virtual Machine. IBM Systems Journal, Java Performance Issue, Vol. 39, No. 1, 2000.

Burke, Choi, Fink, Grove, Hind, Sarkar, Serrano, Sreedhar, Srinivasan, Whaley. The Jalapeno Dynamic Optimizing Compiler for Java. ACM 1999 Java Grande Conference.

Whaley, Rinard. Compositional Pointer and Escape Analysis for Java Programs. Object-Oriented Programming Systems, Languages, and Applications (OOPSLA 1999), Denver, CO.
#93 on Top 10,000 most cited CS articles of 1999.

Other Publications

Whaley, Carbin, Lam. Finding Effective Variable Orderings for BDD-Based Program Analysis. Submitted for publication.

Whaley, Kozyrakis. Heuristics for Dynamic and Profile-Driven Method-Level Speculative Parallelization. Submitted for publication.

Whaley. Dynamic Optimization through the use of Automatic Runtime Specialization. M.Eng. thesis, Massachusetts Institute of Technology, May 1999.

Whaley, Barton. A real-time performance profiler for Java. Dr. Dobbs Journal, March 1998.

Invited Talks and Presentations

- Stanford University Broad-Area Seminar, January 2005. Hot Topics in Compilers and Programming Languages.
- Intel Research, Sunnyvale, CA, November 2004. Program Analysis using Binary Decision Diagrams.
- PLDI, Washington DC, June 2004. Cloning-Based Context-Sensitive Pointer Alias Analyses Using Binary Decision Diagrams.
- Stanford University CS343, June 2004. Combining Predicates and Context-Sensitive Pointers.
- Stanford University CS343, May 2004. 0wn3rship Types.
- Stanford Computer Forum, May 2004. Program Analysis with bddbddb.
- Sun Microsystems, Mountain View, CA, August 2003. Design and Implementation of the Joeq Virtual Machine.
- IVME Workshop, San Diego, CA, June 2003. Joeq: A Virtual Machine and Compiler Infrastructure.
- Stanford University EE392C, April 2003. Emerging Applications: Verification.
- Stanford University CS243, March 2003. Dynamic Compilation.
- SAS, Madrid, Spain. September 2002. An Efficient Inclusion-Based Points-To Analysis for Strictly-Typed Languages.
- ISSTA, Rome, Italy. July 2002. Automatic Extraction of Object-Oriented Component Interfaces.
- Stanford University CS343, May 2002. Context-Sensitive Analysis.
- Stanford University CS343, May 2002. Context-Sensitive Flow Analysis Using Instantiation Constraints.
- Stanford University CS243, January 2002. The Joeq Compiler System.
- Stanford University Seminar, December 2001. An Application for Flow- and Context-Sensitive Pointer Analysis.

- OOPSLA, Tampa, FL, October 2001. Partial Method Compilation using Dynamic Profile Information.
- Reflection, Kyoto, Japan, September 2001. System Checkpointing using Reflection and Program Analysis.
- Stanford University CS343, June 2001. Scalable, Precise Pointer Analysis (and other oxymorons)
- Stanford University CS343, May 2001. Pointer Analysis That Works.
- Stanford University CS343, May 2001. What's so hard about pointers?
- ACM Java Grande, San Francisco, CA, June 2000. A Portable Sampling-Based Profiler for Java Virtual Machines.
- OOPSLA, Denver, CO, November 1999. Compositional Pointer and Escape Analysis for Java Programs.
- University of Tokyo. July 2000. Effectively gathering and using dynamic profile data.
- Tokyo Institute of Technology. July 2000. Analysis and Optimizations of Multithreaded Programs.
- IBM Japan, 1999. December 1999. Pointer and Escape Analysis for Java.
- Tokyo Institute of Technology. December 1999. Automatic Dynamic Specialization.
- University of Tokyo. September 1999. Compositional Pointer and Escape Analysis for Java Programs.
- Silicon Graphics. December 1998. The Makings of a Seamless Java Virtual Machine.
- MIT 6.892, December 1998. Escape Analysis in Java.
- MIT 6.892, November 1998. Jalapeño: The Makings of a Seamless Java Virtual Machine.
- IBM T.J. Watson Research Center, March 1998. Porting IBvM to x86.
- IBM T.J. Watson Research Center, August 1997. Single-Pass Java Bytecode to IR.

Professional Activities

Parallel and Distributed Embedded Systems (PDES) Program Committee, 2005.
Compiler Optimization Meets Compiler Verification (COCV) Program Committee, 2002.
Reviewer for POPL, PLDI, OOPSLA, ECOOP, TOPLAS, FSE, CGO.

Open Source Projects

Joeq virtual machine and compiler infrastructure. http://joeq.sourceforge.net
JavaBDD binary decision diagram library. http://javabdd.sourceforge.net
bddbddb: BDD-Based Deductive DataBase. http://bddbddb.sourceforge.net
Eclipse KeepResident plugin. http://suif.stanford.edu/pub/keepresident
BuDDy BDD package. http://buddy.sourceforge.net

Honors and Awards

Intel Graduate Fellowship Award, 2004-2005.
Stanford Engineering Fellowship, 2003-2004.
NSF Graduate Student Fellowship Award, 1999-2003.
Paper Awards at PLDI 2004, ISSTA 2002, OOPSLA 2001.
Bell Northern Outstanding Undergraduate Research Prize, 1997.
Faculty Letters for outstanding performance in MIT courses:
    Structure and Interpretation of Computer Programs (6.001), Computation Structures (6.004),
    Artificial Intelligence (6.034), Compiler Design (6.035), Software Engineering Lab (6.170)
USA Computer Olympiad '93: Went to national finals as one of the Top 15 programmers in US.

Languages

English (native speaker), Japanese (fluent)

Citizenship

US Citizen

References

Prof. Monica S. Lam
Stanford University
Gates Building, Room 307
Stanford, CA 94305
(650) 725-3714
`lam@stanford.edu`

Dr. John J. Barton
IBM Research
16181 Escobar Avenue
Los Gatos, CA 95032
(408) 358-3388
`bartonjj@us.ibm.com`

Prof. Dawson Engler
Stanford University
Gates Building, Room 314
Stanford, CA 94305
(650) 723-0762
`engler@stanford.edu`

Prof. Martin C. Rinard
Massachusetts Institute of Technology
32 Vassar Street, #32-G866
Cambridge, MA 02139
(617) 258-6922
`rinard@lcs.mit.edu`