

# Matthew Flatt

## Education

- 99    **Ph.D. Computer Science**    Rice University  
Dissertation: *Programming Languages for Reusable Software Components*  
Advisor: Matthias Felleisen
- 98    **M.S. Computer Science**    Rice University
- 93    **B.S. Math and Computer Science, B.S. Physics**    Carnegie Mellon University

## Professional Experience

- 06-    **Associate Professor**    University of Utah
- 00-06    **Assistant Professor**    University of Utah
- 99-00    **Research Assistant Professor**    University of Utah
- 94-99    **Teaching and Research Assistant**    Rice University
- 93-94    **Programmer**    Automation Programming, Inc.
- 89-93    **Programmer**    Carnegie Mellon University

## Publications

### Books

- [1]    10    SPERBER, M., R.K. DYBVIG, M. FLATT, AND A. VAN STRAATEN, EDITORS. Revised<sup>6</sup> Report on the Algorithmic Language Scheme. Cambridge University Press, 301 pages.
- [2]    09    FELLEISEN, M., R.B. FINDLER, AND M. FLATT. *Semantics Engineering with PLT Redex*, MIT Press, 528 pages.
- [3]    01    FELLEISEN, M., R.B. FINDLER, M. FLATT, AND S. KRISHNAMURTHI. *How to Design Programs*, MIT Press, 693 pages.

### Journal Publications and Book Chapters

- [4]    06    FINDLER, R.B. AND M. FLATT. Slideshow: Functional Presentations. *Journal of Functional Programming*, 16(4-5), pages 583–619.
- [5]    04    FELLEISEN, M., R.B. FINDLER, M. FLATT, AND S. KRISHNAMURTHI. The Teach-Scheme! Project: Computing and Programming for Every Student. *Computer Science Education*, 14(1), pages 55–77.
- [6]    04    FELLEISEN, M., R.B. FINDLER, M. FLATT, AND S. KRISHNAMURTHI. The Structure and Interpretation of the Computer Science Curriculum. *Journal of Functional Programming*, 14(4), pages 365–378.
- [7]    02    FINDLER, R.B., J. CLEMENTS, C. FLANAGAN, M. FLATT, S. KRISHNAMURTHI, P. STECKLER, AND M. FELLEISEN. DrScheme: A Programming Environment for Scheme. *Journal of Functional Programming*, 12(2), pages 159–182.
- [8]    99    FLATT, M., S. KRISHNAMURTHI, AND M. FELLEISEN. A Programmer’s Reduction Semantics for Classes and Mixins. In *Formal Syntax and Semantics of Java*, LNCS volume 1523, Springer-Verlag, pages 241–269.

- [9] 93 COHEN, J.D., B. MACWHINNEY, M. FLATT, AND J. PROVOST. PsyScope: A New Graphic Interactive Environment For Designing Psychology Experiments. *Behavioral Research Methods, Instruments, and Computers*, 25(2), pages 257–271.

### Conference Publications

- [10] 10 ATKINSON, K., M. FLATT, AND G. LINDSTROM. Using Macros to Address ABI Compatibility. In *Proc. Conference on Generative Programming and Component Engineering (GPCE)*.
- [11] 10 SWAINE, J., K. TEW, P. DINDA, R.B. FINDLER, AND M. FLATT. Back to the Futures: Incremental Parallelization of Existing Sequential Runtime Systems. In *Proc. ACM Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA)*.
- [12] 10 KLEIN, C., M. FLATT, AND R.B. FINDLER. Random Testing for Higher-Order, Stateful Programs. In *Proc. ACM Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA)*.
- [13] 09 FLATT, M., E. BARZILAY, AND R.B. FINDLER. Scribble: Closing the Book on Ad Hoc Documentation Tools. In *Proc. ACM International Conference on Functional Programming (ICFP)*, pages 109–120. [Conf. acceptance rate: 26/58].
- [14] 09 FELLEISEN, M., R.B. FINDLER, M. FLATT, AND S. KRISHNAMURTHI. A Functional I/O System (or Fun for Freshman Kids). In *Proc. ACM International Conference on Functional Programming (ICFP)*, pages 47–58. [Conf. acceptance rate: 26/58].
- [15] 09 RAFKIND, J., A. WICK, J. REGEHR, AND M. FLATT. Precise Garbage Collection for C. In *Proc. ACM International Symposium on Memory Management (ISMM)*, pages 39–48. [Conf. acceptance rate: 15/32].
- [16] 07 FLATT, M., G. YU, R.B. FINDLER, AND M. FELLEISEN. Adding Delimited and Composable Control to a Production Programming Environment. In *Proc. ACM International Conference on Functional Programming (ICFP)*.
- [17] 06 OWENS, S. AND M. FLATT. From Structures and Functors to Modules and Units. In *Proc. ACM International Conference on Functional Programming (ICFP)*, pages 87–98. [Conf. acceptance rate: 24/76].
- [18] 06 MCDIRRID, S., W. HSIEH, AND M. FLATT. A Framework for Modular Linking in OO Languages. In *Proc. Joint Modular Languages Conference (JMLC)*.
- [19] 05 GRAY, K., R.B. FINDLER AND M. FLATT. Fine-Grained Interoperability through Mirrors and Contracts. In *Proc. ACM Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA)*, pages 231–245. [Conf. acceptance rate: 29/142].
- [20] 05 CULPEPPER, R., S. OWENS, AND M. FLATT. Syntactic Abstraction in Component Interfaces. In *Proc. Conference on Generative Programming and Component Engineering (GPCE)*, pages 373–388.
- [21] 04 GOLDBERG, D., R.B. FINDLER, AND M. FLATT. Super and Inner — Together at Last! In *Proc. ACM Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA)*, pages 116–129. [Conf. acceptance rate: 27/173].
- [22] 04 WICK, A. AND M. FLATT. Memory Accounting without Partitions. In *Proc. ACM International Symposium on Memory Management (ISMM)*, 11 pages. [Conf. acceptance rate: 15/43].
- [23] 04 FINDLER, R.B. AND M. FLATT. Slideshow: Functional Presentations. In *Proc. ACM International Conference on Functional Programming (ICFP)*, pages 224–235. [Conf. acceptance rate: 21/80].
- [24] 04 FLATT, M. AND R.B. FINDLER. Kill-Safe Synchronization Abstractions. In *Proc. ACM Conference on Programming Language Design and Implementation (PLDI)*, pages 47–58. [Conf. acceptance rate: 25/128].

- [25] 04 FINDLER, R.B., M. FLATT, AND M. FELLEISEN. Semantic Casts: Contracts and Structural Subtyping in a Nominal World. In *Proc. European Conference on Object-Oriented Programming (ECOOP)*, pages 364–388. [Conf. acceptance rate: 25/132].
- [26] 04 J. MATTHEWS, R.B. FINDLER, M. FLATT, AND M. FELLEISEN. A Visual Environment for Developing Context-Sensitive Term Rewriting Systems. In *Proc. International Conference on Rewriting Techniques and Applications (RTA)*. [Conf. acceptance rate: 19/43].
- [27] 02 FLATT, M. Composable and Compilable Macros: You Want it *When?* In *Proc. ACM International Conference on Functional Programming (ICFP)*, pages 72-83. [Conf. acceptance rate: 24/76].
- [28] 01 MCDIR MID, S., M. FLATT, AND W. HSIEH. Jiazzzi: New-Age Components for Old-Fashioned Java. In *Proc. ACM Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA)*, pages 211-222. [Conf. acceptance rate: 27/145].
- [29] 01 CLEMENTS, J., M. FLATT, AND M. FELLEISEN. Modeling an Algebraic Stepper. In *Proc. European Symposium on Programming (ESOP)*, pages 320–334. [Conf. acceptance rate: 26/76].
- [30] 00 REID, A., M. FLATT, L. STOLLER, J. LEPREAU, AND E. EIDE. Knit: Component Composition for Systems Software. In *Proc. USENIX Conference on Operating System Design and Implementation (OSDI)*, pages 347–360. [Conf. acceptance rate: 24/111].
- [31] 99 FLATT, M., R.B. FINDLER, S. KRISHNAMURTHI, AND M. FELLEISEN. Programming Languages as Operating Systems (*or Revenge of the Son of the Lisp Machine*). In *Proc. ACM International Conference on Functional Programming (ICFP)*, pages 138–147. [Conf. acceptance rate: 25/81].
- [32] 98 FLATT, M. AND M. FELLEISEN. Units: Cool Modules for HOT Languages. In *Proc. ACM Conference on Programming Language Design and Implementation (PLDI)*, pages 236–248. [Conf. acceptance rate: 31/136].
- [33] 98 FLATT, M., S. KRISHNAMURTHI, AND M. FELLEISEN. Classes and Mixins. In *Proc. ACM Symposium on Principles of Programming Languages (POPL)*, pages 171–183. [Conf. acceptance rate: 32/175].
- [34] 98 FINDLER, R.B. AND M. FLATT. Modular Object-Oriented Programming with Units and Mixins. In *Proc. ACM International Conference on Functional Programming (ICFP)*, pages 94–104. [Conf. acceptance rate: 30/77].
- [35] 97 FINDLER, R.B., C. FLANAGAN, M. FLATT, S. KRISHNAMURTHI, AND M. FELLEISEN. DrScheme: A Pedagogic Programming Environment for Scheme. In *Proc. International Symposium on Programming Languages, Implementations, Logics, and Programs (PLILP)*, pages 369–388. [Conf. acceptance rate: 25/68].
- [36] 96 FLANAGAN, C., M. FLATT, S. KRISHNAMURTHI, S. WEIRICH, AND M. FELLEISEN. Finding Bugs in the Web of Program Invariants. In *Proc. ACM Conference on Programming Language Design and Implementation (PLDI)*, pages 23–32. [Conf. acceptance rate: 28/112].

### Workshop Papers

- [37] 10 BARLAND, I., R.B. FINDLER, AND M. FLATT. The Design of a Functional Image Library. In *Proc. Scheme Workshop*.
- [38] 09 FLATT, M., AND E. BARZILAY Keyword and Optional Arguments in PLT Scheme. In *Proc. Scheme Workshop*.
- [39] 07 CULPEPPER, R., S. TOBIN-HOCHSTADT, AND M. FLATT. Advanced Macrology and the Implementation of Typed Scheme. In *Proc. Scheme Workshop*.
- [40] 04 GRAY, K. AND M. FLATT. Compiling Java to PLT Scheme. In *Proc. Scheme Workshop*.

- [41] 04 OWENS, S., M. FLATT, O. SHIVERS, AND B. MCMULLAN. Lexer and Parser Generators in Scheme. In *Proc. Scheme Workshop*.
- [42] 03 GRAY, K. AND M. FLATT. ProfessorJ: A Gradual Intro to Java through Language Levels. In *OOPSLA Educator's Symposium*.
- [43] 02 WICK, A., M. FLATT, AND W. HSIEH. Reachability-Based Memory Accounting. In *Proc. Scheme Workshop*.
- [44] 02 FELLEISEN, M., R.B. FINDLER, M. FLATT, AND S. KRISHNAMURTHI. The Structure and Interpretation of the Computer Science Curriculum. In *Workshop on Functional and Declarative Programming in Education*.
- [45] 01 MCDIRMIID, S., M. FLATT, AND W. HSIEH. Mixing COP and OOP. In *Workshop on Language Mechanisms for Programming Software Components at OOPSLA*.
- [46] 01 EIDE, E., A. REID, M. FLATT, AND J. LEPREAU. Aspect Weaving as Component Knitting: Separating Concerns with Knit. In *Workshop on Advanced Separation of Concerns in Software Engineering at ICSE*.

### Technical Reports and Reference Manuals

- [47] 97 FLATT, M. *PLT MzScheme: Language Reference Manual*. Rice CS TR97-280, 150 pages. Updated 2005
- [48] 97 FLATT, M., R.B. FINDLER, AND J. CLEMENTS. *PLT MrEd: Graphical Toolbox Manual*. Rice CS TR97-281, 315 pages. Updated 2005
- [49] 96 FLATT, M. *MrEd: An Engine for Portable Graphical User Interfaces*. Rice CS TR96-258, 15 pages.

### Other Publications

- [50] 06 FLATT, M., R.B. FINDLER., AND M. FELLEISEN. Scheme with Classes, Mixins, and Traits (invited tutorial). In *Asian Symposium on Programming Languages and Systems (APLAS)*, pages 270–289.
- [51] 04 FELLEISEN., M., R.B. FINDLER, M. FLATT, AND S. KRISHNAMURTHI. Building Little Languages with Macros. *Dr. Dobbs Journal*, pages 45–49 (April).
- [52] 04 CLEMENTS, J., M. FELLEISEN., R.B. FINDLER, M. FLATT, AND S. KRISHNAMURTHI. Fostering Little Languages. *Dr. Dobbs Journal*, pages 16–24 (March).
- [53] 98 FELLEISEN, M., R.B. FINDLER, M. FLATT, AND S. KRISHNAMURTHI. The DrScheme Project: An Overview. *SIGPLAN Notices*, pages 17–23 (June).
- [54] 97 FLATT, M. AND M. FELLEISEN. Program Units as Higher-Order Modules. In *Proc. Workshop on Higher-Order Operational Techniques in Semantics*, ENTCS volume 10, Elsevier, 26 pages.

### Unpublished

- [55] 03 FELLEISEN, M. AND M. FLATT. *Programming Languages and Lambda Calculi*. Unpublished monograph. Superseded by *Semantics Engineering with PLT Redex*.

### Research Support

- 09-12 MATTHEW FLATT (PI). An Extensible Gradual Type System via Compile-Time Meta-Programming. NSF CCF SHF program, \$419,000.
- 09-11 MATTHIAS FELLEISEN (PI), ROBERT BRUCE FINDLER, MATTHEW FLATT (LOCAL PI), SHRI-RAM KRISHNAMURTHI. Infrastructure for the Production of Languages. NSF CI-ANND-EN, \$134,415 (total for all sites: \$660,000).

- 06-10 STEPHEN BLOCH (PI), JOHN B. CLEMENTS, KATHI FISLER, MATTHEW FLATT (LOCAL PI), VIERA K. PROULX. Redesigning Introductory Computing: The Design Discipline. NSF DUE program, \$20,000 (total for all sites: \$499,688, mostly to support workshop participants across all sites).
- 05-08 MATTHEW FLATT (PI), JOHN REGEHR. Experimenting with Garbage Collection in an Otherwise Conventional Operating System. NSF PDOS program, \$380,000.
- 04-07 MATTHEW FLATT (LOCAL PI), collaborative with OLIN SHIVERS (LEAD PI) (GATech), PETE MANOLIOS (GATech). Language Towers as Design Frameworks. NSF SOD program, \$180,000 (total for all sites: \$630,000).
- 04-07 MATTHEW FLATT (LOCAL PI), JOHN REGEHR, ERIC EIDE, collaborative with RAY KLEFSTAD (LEAD PI) (UCI). Collaborative Research: EHS: Components and Aspects for Embedded Middleware. NSF EHS program, \$360,000 (total for all sites: \$630,000). *This proposal was submitted by Wilson Hsieh, Eric Eide, and Ray Klefsted.*
- 03-05 MATTHEW FLATT (LOCAL PI), collaborative with ROBERT BRUCE FINDLER (LEAD PI) (UChicago). Collaborative: Exploiting Component Contracts for Static Analysis and Testing. NSF SEL program, \$159,987 (total for all sites: \$270,000).
- 03 MATTHEW FLATT (PI). PLT Scheme for VCC. DARPA seedling contract, \$53,950.
- 01-04 MATTHIAS FELLEISEN (PI) (Northeastern), STEPHEN BLOCH (Gustavus Adolphus), KATHI FISLER (WPI), MATTHEW FLATT, SHRIRAM KRISHNAMURTHI (Brown). Computing Education for Every Student in Secondary School. NSF ESI program, \$184,236 (total for all sites: \$1,606,205).
- 00-04 JAY LEPREAU (PI) AND MATTHEW FLATT. Alchemy: Transmuting Raw Components into Robust Code. DARPA PCES, \$2,900,000.

## Software

- PLT Scheme** with Robert Bruce Findler et al. <http://www.plt-scheme.org/>  
95-present PLT Scheme is a suite of programming tools that includes DrScheme, a graphical programming environment that runs under Windows, Mac OS, and Unix/X.

*Average downloads/day (unique IPs using main site) as of September 2004: 175*  
*Subscribers to plt-scheme user mailing list as of September 2004: 551*  
*Average messages/day to plt-scheme list during September 2004: 7.7*

- PsyScope** with Jefferson Provost <http://psyscope.psy.cmu.edu/>  
90-94 PsyScope is a system for creating and running experiments in cognitive psychology on Macintosh computers.

*A consortium of 37 labs ported PsyScope to Mac OS X in 2004: <http://psy.ck.sissa.it>*

## Invited Talks and Instruction

09	Colloquium	Brigham Young University	Provo, UT
06	Invited Tutorial	Asian Sym. on Prog. Lang. and Systems (APLAS)	Sydney, Australia
06	Invited Lectures	Oregon Summer School on Concurrency	Eugene, OR
04	Colloquium	University of Chicago	Chicago, IL
04	Colloquium	University of Washington	Seattle, WA
04	Colloquium	Microsoft Research	Redmond, WA
03	Invited Mini-Tutorial	Conf. on Generative Programming and Component Engineering (GPCE)	Erfurt, Germany
03	Panel, $\frac{1}{2}$ -Day Tutorial	SIGCSE Conference	Reno, NV

03	Invited Lecture	Semana de Computación en Ciencias, National University of Mexico	Mexico City
02	Submitted Presentation	Lightweight Languages Workshop	Boston, MA
02	Colloquium	Georgia Institute of Technology	Atlanta, GA
02	Colloquium	Universität Tübingen	Tübingen, Germany
02	1-Day Tutorial	SIGCSE Conference	Cincinnati, OH
02	Semantics Course Lecture	Brown University	Providence, RI
02	Programming Languages Seminar	Yale University	New Haven, CT
01	Programming Languages Seminar	Northeastern University	Boston, MA
01	Systems Seminar	Purdue University	West Lafayette, IN
01	Colloquium	South China University of Technology	Canton, China

## Teaching

F09	Instructor for CS 5510/6510, Operating Systems Enrollment — 61 students	University of Utah
F07-09	Instructor for CS 5510, Programming Languages Enrollment — F09: 18; F07: 27 students	University of Utah
S00-04, S06, S09	Instructor for CS 6520/7520, Programming Languages and Semantics Enrollment — S09: 15; S06: 15; S04: 11; S03: 11; S02: 27; S01: 19; S00: 17 students	University of Utah
F08	Instructor for CS 4960-01, Parallel Programming Enrollment — 31 students	University of Utah
S08	Instructor for CS 6510, Functional Programming Enrollment — 14 students	University of Utah
S08	Instructor for CS 5967, How to Design Programs Enrollment — 4 students	University of Utah
F00-02, F05	Instructor for CS 3520, Programming Language Concepts Enrollment — F05: 20; F02: 44; F01: 82; F00: 73 students	University of Utah
S05	Instructor for CS 7963, Practical Functional Programming Enrollment — 10 students	University of Utah
F03	Instructor for CS 2010, Computer Science I Enrollment — 136 students	University of Utah
97-99	Instructor for <i>TeachScheme!</i> Workshop	Rice University
S96	Teaching assistant for COMP 311, Programming Languages	Rice University
F94, S95, F95	Lab instructor for COMP 210, Intro Computer Science	Rice University

## Curriculum Development

### TeachScheme!

The *TeachScheme!* curriculum is used in introductory computer science courses at the high school through university levels. The DrScheme programming environment and *How to Design Programs* book are part of this curriculum.  
<http://www.htdp.org/>

07-09	Directed summer <i>TeachScheme/ReachJava</i> workshop	University of Utah
00-04	Directed summer <i>TeachScheme!</i> workshop for high-school teachers	University of Utah
02	Directed <i>TeachScheme!</i> workshop for faculty/staff/TAs	Georgia Tech

### Semantics of Programming Languages

Created new course notes [55] for CS 6520 to provide a concise and up-to-date presentation of semantic techniques for current languages research.

## Thesis Committees

07-	Kevin Tew		PhD thesis committee	Chair/Advisor
07-	Jon Rafkind		PhD thesis committee	Chair/Advisor
06-09	Eric Eide		PhD thesis committee	Chair/Advisor
05-	Kevin Atkinson		PhD thesis committee	Co-Chair/Co-Advisor
00-06	Kathy Gray		PhD thesis committee	Chair/Advisor
00-06	Scott Owens		PhD thesis committee	Co-Chair/Co-Advisor
00-06	Adam Wick		PhD thesis committee	Chair/Advisor
07-	Xuejun Yang		PhD thesis committee	Member
07-	Jianjun Duan		PhD thesis committee	Member
07-	Will Archer		PhD thesis committee	Member
07-10	Guodong Li		PhD thesis committee	Member
07-10	Ryan Culpepper	Northeastern	PhD thesis committee	External Member
07-08	Nathan Coopridner		PhD thesis committee	Member
04-07	Nicole Anderson		PhD thesis committee	Member
03-06	Kostadin Damevski		PhD thesis committee	Member
03-05	Sean McDirmid		PhD thesis committee	Member
02-04	Ali Sezgin		PhD thesis committee	Member
01-05	John Clements	Northeastern Univ.	PhD thesis committee	External Member
01-04	Jason Yang		PhD thesis committee	Member
00-07	Peter Jensen		PhD thesis committee	Member
09-10	Gabe Rudy		MS thesis committee	Member
09-09	James Swaine	Northwestern	MS thesis committee	External Member
08-09	Casey Klein	U. Chicago	MS thesis committee	External Member
02-04	Mac Newbold		MS thesis committee	Member
02-03	Kostadin Damevski		MS thesis committee	Member
01-02	William Phillips		MS thesis committee	Member
01-02	Brian Remick		MS thesis committee	Member
99-01	Jason Baker		MS thesis committee	Member

## Other Advising

06	Yun Jin Kang	Undgraduate Research (UROP)	Advisor
----	--------------	-----------------------------	---------

## Service

### Steering Committees

04-07	ACM Intl. Conf. on Functional Programming (ICFP)	Member
02-04	Scheme Workshop	Member

### Program Committees

12	Intl. Conf. on Compiler Construction (CC)	Member
----	---	--------

11	ACM Conf. on Obj-Oriented Prog, Langs, Sys, and Apps (OOPSLA)	Member
10	Workshop on Dynamic Languages and Applications (Dyla)	Member
09	Sym. on Implementation and Application of Functional Languages (IFL)	Member
09	Intl. Conf. on Software Composition (SC)	Member
09	ACM Conf. on Principles of Programming Languages (POPL)	Member
08	Workshop on Virtual Machines and Intermediate Languages (VMIL)	Member
08	Functional and Declarative Programming in Education (FDPE)	Member
08	Asian Symposium on Programming Languages and Systems (APLAS)	Member
08	Sym. on Practical Aspects of Declarative Languages (PADL)	Member
08	ACM Workshop on Foundations of Object-Oriented Languages (FOOL)	Member
07	Workshop on Dynamic Languages and Applications (Dyla)	Member
06	ACM Conf. on Programming Language Design and Implementation (PLDI)	Member
06	Sym. on Practical Aspects of Declarative Languages (PADL)	Member
06	Workshop on Foundations of Aspect-Oriented Languages (FOAL)	Member
05	ACM Dynamic Languages Symposium (DLS)	Member
05	ACM Intl. Conf. on Functional Programming (ICFP)	Member
04	Conf. on Generative Programming & Component Engineering (GPCE)	Member
03	Scheme Workshop	Chair
03	Workshop on Object-Oriented Developments (WOOD)	Member
03, 04	Sym. on Practical Aspects of Declarative Languages (PADL)	Member
02	ACM Intl. Conf. on Functional Programming (ICFP)	Member
01	Scheme Workshop	Member

### Other Committees

09-10	ACM SIGPLAN Outstanding Doctoral Dissertation Award Committee	Chair
09-12	ACM SIGPLAN Executive Committee	Member-at-large
05	ICFP Programming Contest organizers	Member
04-07	Scheme Language Standard (R6RS) editors	Member

### Journal Guest Editing

06	<i>J. Functional Programming</i> , Special Issue on ICFP'05	Guest Co-Editor
----	---	-----------------

### Journal Editorial Board

09-11	<i>J. Functional Programming</i>	Member
-------	----------------------------------	--------

### Reviewing

Reviewer for *Transactions on Programming Languages and Systems* (×5), *IEEE Transactions on Software Engineering* (×2), *Computing Surveys*, *Journal of Functional Programming* (×6), *Theory and Practice of Object Systems*, *Software: Practice & Experience* (×3), *Higher-Order and Symbolic Computation* (×3), *Computer Languages, Systems, and Structures* (×2), *Science of Computing, Information and Software Technology*.

External reviewer for ICFP'10, ECOOP'09, OOPSLA'08, ECOOP'08, PLDI'08, ASOD'08, ESOP'08, POPL'08, OOPSLA'07, PADL'07, POPL'07, SCHEME'06, ICALP'06, USENIX'06, POPL'06, ICALP'05, ICFP'04, ESOP'04, TYPES'03, ICFP'03, POPL'03, ECOOP'02, ICFP'01, PLDI'01, POPL'01, ICFP'00, POPL'00, ICFP'99, PLDI'99, PPDP'99, OOPSLA'98, and ICFP'96.

NSF proposal review panel: 05, 06.

**College of Engineering**

05-06	College Curriculum Committee	Member
04-06	College Council	Member

**School of Computing**

08-10	Industry Forum	Organizer
08-10	Graduate Admissions	Member
05	Facility Task Force	Member
04-05	Facility Committee	Member
03-06	Curriculum Committee	Member
02-03	Distinguished Lecture Series	Organizer
00-03	Undergraduate Committee	Member
00-03	Foundations Comprehensive Exam Committee	Member

**Awards**

09	Outstanding Teaching Award	School of Computing, University of Utah
05	Early Career Teaching Award	University of Utah