Antony L. Hosking

Curriculum vitæ

Computer Science, Purdue University
305 N. University St., W. Lafayette, IN 47907
+1 765 427 5484 (mobile)
+1 317 255 4485 (home)
hosking@cs.purdue.edu
www.cs.purdue.edu/~hosking
July 22, 2015

Personal information

Citizenship Australian (descent), Canadian (birth), USA (naturalized)

Education

1995 **PhD**, *Computer Science*, University of Massachusetts, Amherst.
Dissertation: *Lightweight support for fine-grained persistence on stock hardware.*

1987 **MSc (1st Class Honours)**, *Computer Science*, University of Waikato, Hamilton, New Zealand.
Thesis: *A semantic algebra approach to denotational semantics of programming languages.*

1985 **BSc**, *Mathematical Sciences*, University of Adelaide, Australia.

Research interests

Programming language design and implementation: compilers, interpreters, and optimization; run-time systems; concurrent and real-time garbage collection; security and safety. Object-oriented database systems; database and persistent programming languages and systems. Architectural support for programming languages and applications. Experimental performance evaluation of prototype systems.

Honors

2014 **ACM Recognition of Service Award**.
For contributions to ACM as General Chair of SPLASH 2013.

2014 **Engagement Award**, *College of Science*, Purdue University, West Lafayette.
For organization of SPLASH 2013 as General Chair.

2013 **Member**, *AITO (Association Internationale pour les Technologies Objets)*.
The current AITO membership numbers 45 persons, world-wide.

2012 **ACM Distinguished Scientist**, *Association for Computing Machinery*.
Recognizes up to 10% of the top ACM members, with at least 15 years of professional experience that had significant accomplishments or impact in the computing field. The grade has three categories: Distinguished Educator, Distinguished Engineer and Distinguished Scientist, recognizing achievements in different areas. Nominations for each category are considered separately.

2009 **ACM Recognition of Service Award**.
For contributions to ACM as General Chair of VEE'09.

2008 **ACM Senior Member**, *Association for Computing Machinery*.
Recognizes up to 25% of the top ACM Professional Members with at least 10 years of professional experience and 5 years of continuous Professional Membership who have demonstrated performance that sets them apart from their peers.

2006 **ACM Recognition of Service Award**.
For contributions to ACM as General Chair of MSPC'06.

2003 **Top 10 Teacher Award**, *College of Science*, Purdue University, West Lafayette.
Recognizes teachers nominated by students as among the top 10 teachers within the college.

2000 **ACM Recognition of Service Award**.
For contributions to ACM as Program Chair of ISMM'00.

1998 **Teaching for Tomorrow Award**, Purdue University, West Lafayette.
Recognizes eight faculty campus-wide for their outstanding teaching potential.

Professional experience

from August 2015 **Professor**, *Research School of Computer Science*, The Australian National University, Canberra.
(fixed term appointment with 50% research contribution to NICTA Australia)

since June 2001 **Associate Professor**, *Department of Computer Science*, Purdue University, West Lafayette, Indiana.
(tenured, on research leave from August 2015)

January–July 2013 **Visiting Senior Principal Researcher**, *NICTA Australia*, Sydney, Australia.

December 2008–March 2015 **Visiting Fellow**, *Research School of Computer Science*, The Australian National University, Canberra, Australia.

May–November 2008 **Visiting Researcher**, *School of Computing*, University of Kent, Canterbury, UK.
Funded by UK Engineering and Physical Sciences Research Council.

November 2001 **Visiting Researcher**, *School of Computer Science*, University of Adelaide, Australia.

September 2001 **Visiting Researcher**, *School of Computing Science*, University of Glasgow, UK.

January 1995–June 2001 **Assistant Professor**, *Department of Computer Science*, Purdue University, West Lafayette, Indiana.
(tenure-track)

September 1986–December 1994 **Research Assistant**, *Department of Computer Science*, University of Massachusetts, Amherst.

Consulting experience

2013–2014 **Expert witness (reports, deposition)**, *Kilpatrick Townsend*, for Oracle, San Francisco, California.
Thought Inc. v. Oracle Corp., Case No. C12-05601 JWS (California Northern)

2013–2014 **Expert witness (reports, deposition)**, *Perkins Coie LLP*, for Microsoft, San Diego, California.
Enfish LLC v. Microsoft Corporation, Inc., Case No. 12-cv-7460-MRP (California Central)

2011–2012 **Expert witness (reports)**, *Woodcock Washburn LLP*, for Microsoft and SAP, Philadelphia, Pennsylvania.
Microsoft Corporation v. DataTern, Inc., Case No. 11-2365 (New York Southern)
SAP AG and SAP America, Inc. v. DataTern, Inc., Case No. 11-2648 (New York Southern)

2011 **Technical consultant**, *Quinn Emanuel Urquhart & Sullivan LLP*, for Symantec, San Francisco, California.
Finjan v. Symantec, Case No. 10-593 (Delaware)

2007–2010 **Expert witness (reports, deposition) and technical consultant**, *Shore Chan Bragalone LLP*, for Software Tree LLC, Dallas, Texas.
Software Tree v. Red Hat, Case No. 09-97 (Texas Eastern)
Software Tree v. Oracle, Case No. 08-126 (Texas Eastern)

2008–2009 **Technical consultant**, *Kelley Drye & Warren LLP*, for LimitNone LLC, Chicago, Illinois.
LimitNone v. Google, Case No. 08-4178 (Illinois Northern)

2006–2007 **Expert witness**, *Howrey LLP*, for Sun Microsystems, Inc., Washington, DC.
Azul Systems v. Sun Microsystems, Case No. 06-1988 CRB (JCS) (California Northern)

Other experience

2012 **Material witness (deposed)**.
Gemalto S.A. v. HTC Corp. et al., C.A. 6:10-cv-561(LED) (E.D. Tex.) (uncompensated)

Professional society affiliations

since 2012 **ACM Distinguished Scientist**, *Association for Computing Machinery*.
Life member, and member of SIGOPS, SIGARCH, SIGPLAN, and SIGMOD.

2009–2012 **Member-at-large**, *ACM SIGPLAN Executive Committee*, elected by SIGPLAN membership, and subcommittee Chair for ACM SIGPLAN Most-Influential Paper Awards.

2008-2012 **ACM Senior Member**, *Association for Computing Machinery*.
Life member, and member of SIGOPS, SIGARCH, SIGPLAN, and SIGMOD.

1987–2008 **ACM Member**, *Association for Computing Machinery*.
Member of SIGOPS, SIGARCH, SIGPLAN, and SIGMOD.

since 1987 **Member**, *Institute for Electrical and Electronics Engineers (IEEE)*.
Member of IEEE Computer Society.

elected 2001 **Faculty Member**, *Upsilon Pi Epsilon*, the International Computer Science Honor Society, Purdue University chapter.

Publications

with Digital Object Identifiers (DOI) where available

Rigorously refereed articles are separated into journal and conference articles. This does not imply that journal articles are any more rigorously refereed than conference articles, nor that they are more significant in their impact. Citation counts obtained from Google Scholar and listed below (as of 2015-07-21) may aid in assessing impact.

Scholarly books

2011 1. Richard Jones, Antony Hosking, Eliot Moss. *The Garbage Collection Handbook: The Art of Dynamic Memory Management*, 512pp. Chapman & Hall/CRC Applied Algorithms and Data Structures Series, published August 2011, ©2012. ISBN-13: 978-1420082791. `http://www.gchandbook.org` [cited by 92]

Refereed journal articles

2011 2. Kalibera T, Pizlo F, Hosking AL, Vitek J. Scheduling real-time garbage collection on uniprocessors. *ACM Transactions on Computer Systems 3*(1), pages 8:1–29, August 2011.
doi: `10.1145/2003690.2003692` [cited by 7]

3. McGachey P, Hosking AL, Moss JEB. Class transformations for transparent distribution of Java applications. *Journal of Object Technology 10*, pages 9:1–35, August 2011.
doi: `10.5381/jot.2011.10.1.a9` [cited by 3]

2009 4. McGachey P, Hosking AL, Moss JEB. Pervasive load-time transformation for transparently distributed Java. *Electronic Notes in Theoretical Computer Science 253*(5), pages 47–64, December 2009.
doi: `10.1016/j.entcs.2009.11.014` [cited by 6]

2008 5. Blackburn SM, McKinley KS, Garner R, Hoffmann C, Khan AM, Bentzur R, Diwan A, Feinberg D, Frampton D, Guyer SZ, Hirzel M, Hosking A, Jump M, Lee H, Moss JEB, Phansalkar A, Stefanović D, VanDrunen T, von Dincklage D, Wiedermann B. *Research Highlight, by invitation*. Wake up and smell the coffee: evaluation methodology for the 21st century. *Communications of the ACM 51*(8), pages 83–89, August 2008.
doi: `10.1145/1378704.1378723` [cited by 73]

2006 6. Moss JEB, Hosking AL. Nested transactional memory: model and architecture sketches. *Science of Computer Programming 63*, pages 186–201, December 2006.
doi: `10.1016/j.scico.2006.05.010` [cited by 131]

7. Welc A, Jagannathan S, Hosking AL. Revocation techniques for Java concurrency. *Concurrency and Computation — Practice and Experience 18*(2), pages 1613–1656, October 2006.
doi: `10.1002/cpe.1008` [cited by 9]

2005 8. Jagannathan S, Vitek J, Welc A, Hosking A. A transactional object calculus. *Science of Computer Programming 57*(2), pages 164–186, August 2005.
doi: `10.1016/j.scico.2005.03.001` [cited by 52]

2004 9. VanDrunen T, Hosking AL. Anticipation-based partial redundancy elimination for static single assignment form. *Software — Practice and Experience 34*(15), pages 1413–1439, December 2004.

doi: 10.1002/spe.618 [cited by 10]

2001 10. Hosking AL, Nystrom N, Whitlock D, Cutts Q, Diwan A. Partial redundancy elimination for access path expressions. *Software — Practice and Experience 31*(6), pages 577–600, May 2001.
doi: 10.1002/spe.371 [cited by 40]

Refereed conference papers

2015 11. Gammie P, Hosking AL, Engelhardt K. Relaxing Safely: Verification of on-the-fly garbage collection for x86-TSO. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)* (Portland, Oregon, June 2015), pages 99–109.
doi: 10.1145/2737924.2738006 [acceptance rate 58/303=19%]

12. Lin Y, Wang K, Blackburn SM, Hosking AL, Norrish M. Stop and Go: Understanding yieldpoint behavior. *Proceedings of the ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Portland, Oregon, June 2015), pages 70–80 .
doi: 10.1145/2754169.2754187 [acceptance rate 12/25=48%, cited by 1]

13. Hussein A, Hosking AL, Payer M, Vick C. Don't race the memory bus: Taming the GC leadfoot. *Proceedings of the ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Portland, Oregon, June 2015), pages 15–27.
doi: 10.1145/2754169.2754182 [acceptance rate 12/25=48%]

14. Hussein A, Payer M, Hosking AL, Vick C. Impact of GC Design on Power and Performance for Android. *Proceedings of the ACM International Conference on Systems and Storage (SYSTOR)* (Haifa, Israel, May 2015), pages 13:1–12.
doi: 10.1145/2757667.2757674 [acceptance rate 18/51=33%]

15. Wang K, Lin Y, Blackburn SM, Norrish M, Hosking AL. Draining the Swamp: Micro virtual machines as solid foundation for language development. *Proceedings of the Inaugural Summit on Advances in Programming Languages (SNAPL)* (Asilomar, California, May 2015).
doi: 10.4230/LIPIcs.SNAPL.2015.321

2014 16. Chapman K, Hosking AL, Moss JEB, Richards T. Closed and open nested atomic actions for Java: language design and prototype implementation. *Proceedings of the International Conference on the Principles and Practice of Programming on the Java platform: virtual machines, languages, and tools (PPPJ)* (Kraków, Poland, September 2014), pages 169–180.
doi: 10.1145/2647508.2647525 [acceptance rate 15/38=39%]

2012 17. Yang X, Blackburn SM, Frampton D, Hosking AL. Barriers reconsidered, friendlier still! *Proceedings of the ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Beijing, China, June 2012), pages 37–48.
doi: 10.1145/2258996.2259004 [acceptance rate 12/30=40%, cited by 15]

2011 18. Pizlo F, Frampton D, Hosking AL. Fine-grained adaptive biased locking. *Proceedings of the International Conference on the Principles and Practice of Programming on the Java platform: virtual machines, languages, and tools (PPPJ)* (Kongens Lyngby, Denmark, August 2011), pages 171–181.
doi: 10.1145/2093157.2093184 [acceptance rate 17/35=49%, cited by 6]

19. Chapman K, Hussein A, Hosking AL. X10 on the Single-chip Cloud Computer. *Proceedings of the X10 Workshop (X10)* (San Jose, California, June 2011), pages

7:1–8.
doi: 10.1145/2212736.2212743 [cited by 6]

2010 20. Pizlo F, Ziarek L, Maj P, Hosking AL, Blanton E, Vitek J. Schism: fragmentation-tolerant real-time garbage collection. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)* (Toronto, Canada, June 2010), pages 146–159.
doi: 10.1145/1806596.1806615 [acceptance rate 41/206=20%, cited by 50]

21. Hellyer L, Jones R, Hosking AL. The locality of concurrent write barriers. *Proceedings of the ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Toronto, Canada, June 2010), pages 83–92.
doi: 10.1145/1806651.1806666 [acceptance rate 13/30=43%, cited by 3]

2009 22. Kalibera T, Pizlo F, Hosking AL, Vitek J. Scheduling hard real-time garbage collection. *Proceedings of the 30th IEEE Real-Time Systems Symposium (RTSS)* (Washington, DC, December 2009), pages 81–92.
doi: 10.1109/RTSS.2009.40 [acceptance rate 44/197=21%, cited by 20]

23. McGachey P, Hosking AL, Moss JEB. Classifying Java class transformations for pervasive virtualized access. *Proceedings of the ACM SIGPLAN Conference on Generative Programming and Component Engineering (GPCE)* (Denver, Colorado, October 2009), pages 75–84.
doi: 10.1145/1621607.1621620 [acceptance rate 18/62=29%, cited by 4]

24. McGachey P, Hosking AL, Moss JEB. Pervasive load-time transformation for transparently distributed Java. *Proceedings of the 4th International Workshop on Bytecode Semantics, Verification, Analysis and Transformation (BYTECODE)* (York, England, March 2009). [cited by 6]

25. Hambrusch SE, Hoffmann C, Korb JT, Haugan M, Hosking AL. A multidisciplinary approach towards computational thinking for science majors. *Proceedings of the ACM SIGCSE Technical Symposium on Computer Science Education (SIGCSE)* (Chattanooga, Tennessee, March 2009), pages 183–187.
doi: 10.1145/1508865.1508931 [acceptance rate 100/302=33%, cited by 65]

2007 26. Pizlo F, Hosking AL, Vitek J. Hierarchical real-time garbage collection. *Proceedings of the ACM SIGPLAN/SIGBED conference on Languages, Compilers, and Tools for Embedded Systems (LCTES)* (San Diego, California, June 2007), pages 123–133.
doi: 10.1145/1254766.1254784 [acceptance rate 21/76=28%, cited by 14]

27. Ni Y, Menon V, Adl-Tabatabai A, Hosking AL, Hudson RL, Moss JEB, Saha B, Shpeisman T. Open nesting in software transactional memory. *Proceedings of the ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP)* (San Jose, California, March 2007), pages 68–78.
doi: 10.1145/1229428.1229442 [acceptance rate 22/65=34%, cited by 171]

2006 28. Blackburn SM, Garner R, Hoffmann C, Khan AM, McKinley KS, Bentzur R, Diwan A, Feinberg D, Frampton D, Guyer S, Hirzel M, Hosking AL, Jump M, Lee H, Moss JEB, Phansalkar A, Stefanović D, VanDrunen T, von Dincklage D, Wiedermann B. The DaCapo benchmarks: Java benchmarking development and analysis. *Proceedings of the 21st ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (Portland, Oregon, October 2006), pages 169–190.
doi: 10.1145/1167473.1167488 [acceptance rate 26/157=17%, cited by 893]

29. Welc A, Hosking AL, Jagannathan S. Transparently reconciling transactions with locking for Java synchronization. *Proceedings of the 20th European Conference on Object-Oriented Programming (ECOOP)* (Nantes, France, July 2006), *Lecture Notes in Computer Science 4067*, pages 148–173, Springer-Verlag, 2006.
doi: 10.1007/11785477_8 [acceptance rate 21/164=13%, cited by 38]
30. Hosking AL. Portable, mostly-concurrent, mostly-copying garbage collection for multi-processors. *Proceedings of the 4th ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Ottawa, Canada, June 2006), pages 40–51.
doi: 10.1145/1133956.1133963 [acceptance rate 17/45=38%, cited by 7]
31. McGachey P, Hosking AL. Reducing generational copy reserve overhead with fallback compaction. *Proceedings of the 4th ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Ottawa, Canada, June 2006), pages 17–28, ACM, 2006.
doi: 10.1145/1133956.1133960 [acceptance rate 17/45=38%, cited by 24]

2005 32. Moss, JEB, Hosking AL. Nested transactional memory: model and preliminary architecture sketches. *Workshop on Synchronization and Concurrency in Object-Oriented Languages (SCOOL)*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (San Diego, California, October 2005), pages 39–48.
doi: 1802/2099 [acceptance rate 13/23=56%, cited by 65]
33. Welc A, Jagannathan S, Hosking AL. Safe futures for Java. *Proceedings of the 20th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (San Diego, California, October 2005), pages 439–453.
doi: 10.1145/1094811.1094845 [acceptance rate 32/174=18%, cited by 165]

2004 34. Cher C-Y, Hosking AL, Vijaykumar TN. Software prefetching for mark-sweep garbage collection: Hardware analysis and software redesign. *Proceedings of the 11th ACM Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)* (Boston, Massachusetts, October 2004), pages 199–210.
doi: 10.1145/1024393.1024417 [acceptance rate 24/116=14%, cited by 26]
35. Blackburn S, Hosking AL. Barriers: friend or foe? *Proceedings of the 3rd ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Vancouver, Canada, October 2004), pages 143–151, 2004.
doi: 10.1145/1029873.1029891 [acceptance rate 15/43=35%, cited by 76]
36. Welc A, Hosking AL, Jagannathan S. Preemption-based avoidance of priority inversion for Java. *Proceedings of the IEEE International Conference on Parallel Processing (ICPP)* (Montréal, Canada, August 2004), pages 529–538, IEEE, 2004.
doi: 10.1109/ICPP.2004.1327963 [acceptance rate 65/190=34%, cited by 20]
37. Welc A, Jagannathan S, Hosking AL. Transactional monitors for concurrent objects. *Proceedings of the 18th European Conference on Object-Oriented Programming (ECOOP)* (Oslo, Norway, June 2004), *Lecture Notes in Computer Science 3086*, pages 519–542, Springer-Verlag, 2004.
doi: 10.1007/b98195 [acceptance rate 25/132=19%, cited by 107]
38. Vitek J, Jagannathan S, Welc A, Hosking AL. A semantic framework for designer transactions. *Proceedings of the European Symposium on Programming (ESOP), European Joint Conferences on Theory and Practice of Software (ETAPS)* (Barcelona,

Spain, March/April 2004), *Lecture Notes in Computer Science 2986*, pages 249–263, Springer-Verlag, 2004.
doi: 10.1007/b96702 [acceptance rate 27/118=23%, cited by 40]

39. VanDrunen T, Hosking AL. Value-based partial redundancy elimination. *Proceedings of the International Conference on Compiler Construction (CC), European Joint Conferences on Theory and Practice of Software (ETAPS)* (Barcelona, Spain, March/April 2004), *Lecture Notes in Computer Science 2985*, pages 167–184, Springer-Verlag, 2004.
doi: 10.1007/b95956 [acceptance rate 19/58=33%, cited by 22]

2003 40. Mueller F, Hosking AL. Penumbra: An Eclipse plugin for introductory programming. *Proceedings of the Workshop on Eclipse Technology eXchange (eTX)*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (Anaheim, California, Oct 2003), pages 65–68, ACM, 2003.
doi: 10.1145/965660.965674 [cited by 34]

2001 41. Hirzel M, Diwan A, Hosking AL. On the usefulness of liveness for garbage collection and leak detection. *Proceedings of the 15th European Conference on Object-Oriented Programming (ECOOP)* (Budapest, Hungary, June 2001), *Lecture Notes in Computer Science 2072*, pages 181–206, Springer-Verlag, 2001.
doi: 10.1007/3-540-45337-7_11 [acceptance rate 18/108=17%, cited by 17]

42. Whitlock D, Hosking AL. A framework for persistence-enabled optimization of Java object stores. *Proceedings of the 9th International Workshop on Persistent Object Systems (POS)* (Lillehammer, Norway, September 2000), Kirby (Ed), *Lecture Notes in Computer Science 2135*, pages 4–18, Springer-Verlag, 2001.
doi: 10.1007/3-540-45498-5_2 [cited by 1]

1999 43. Hosking AL, Chen J. Mostly-copying reachability-based orthogonal persistence. *Proceedings of the 14th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (Denver, Colorado, November 1999), pages 383–398.
doi: 10.1145/320384.320427 [acceptance rate 30/152=20%, cited by 15]

44. Hosking AL, Chen J. PM3: an orthogonally persistent systems programming language — design, implementation, performance. *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB)* (Edinburgh, Scotland, September 1999), pages 587–598, Morgan Kaufmann, 1999. http://www.vldb.org/conf/1999/P55.pdf [cited by 16]

1998 45. Brahnmath K, Nystrom N, Hosking AL, Cutts Q. Swizzle barrier optimizations for orthogonal persistence in Java. *Proceedings of the 3rd International Workshop on Persistence and Java (PJ)* (Tiburon, California, August 1998), Morrison, Jordan and Atkinson (Eds), *Advances in Persistent Object Systems*, pages 268–278, Morgan Kaufmann, 1999. [cited by 17]

46. Hosking AL, Nystrom N, Cutts Q, Brahnmath K. Optimizing the read and write barriers for orthogonal persistence. *Proceedings of the 8th International Workshop on Persistent Object Systems (POS)* (Tiburon, California, August 1998), Morrison, Jordan and Atkinson (Eds), *Advances in Persistent Object Systems*, pages 149–159, Morgan Kaufmann, 1999. [acceptance rate 16/27=59%, cited by 30]

47. Cutts QI, Lennon S, Hosking AL. Reconciling buffer management with persis-

tence optimizations. *Proceedings of the 8th International Workshop on Persistent Object Systems (POS)* (Tiburon, California, August 1998), Morrison, Jordan and Atkinson (Eds), *Advances in Persistent Object Systems*, pages 51–63, Morgan Kaufmann, 1999. [acceptance rate 16/27=59%, cited by 9]

1997 48. Cutts Q, Hosking AL. Analysing, profiling and optimising orthogonal persistence for Java. *Proceedings of the 2nd International Workshop on Persistence and Java (PJ)* (Half Moon Bay, California, August 1997), Atkinson and Jordan (Eds), *Sun Microsystems Technical Report 97-63*, pages 107–115, December, 1997. `http://research.sun.com/techrep/1997/abstract-63.html` [cited by 14]

1996 49. Hosking AL. Residency check elimination for object-oriented persistent languages. *Proceedings of the 7th International Workshop on Persistent Object Systems (POS)* (Cape May, New Jersey, May 1996), Connor and Nettles (Eds), *Persistent Object Systems: Principles and Practice*, pages 174–183, Morgan Kaufmann, 1997. [acceptance rate 22/50=44%, cited by 11]

50. Moss JEB, Hosking AL. Approaches to adding persistence to Java. *Proceedings of the 1st International Workshop on Persistence and Java (PJ)* (Drymen, Scotland, September 1996), Atkinson and Jordan (Eds), *Sun Microsystems Technical Report 96-58*, pages 1–6, November, 1996. `http://research.sun.com/techrep/1996/abstract-58.html` [cited by 38]

1995 51. Moss JEB, Hosking AL. Expressing object residency optimizations using pointer type annotations. *Proceedings of the 6th International Workshop on Persistent Object Systems (POS)* (Tarascon, France, September 1994), Atkinson, Maier and Benzaken (Eds), *Persistent Object Systems*, pages 3–15, Springer-Verlag, 1995. [acceptance rate 29/54=53%, cited by 21]

1993 52. Hosking AL, Moss JEB. Protection traps and alternatives for memory management of an object-oriented language. *Proceedings of the 14th ACM Symposium on Operating Systems Principles (SOSP)* (Asheville, NC, December 1993), *ACM Operating Systems Review* 27(5), pages 106–119, December 1993. doi: `10.1145/168619.168628` [acceptance rate 21/78=27%, cited by 73]

53. Hosking AL, Moss JEB. Object fault handling for persistent programming languages: A performance evaluation. *Proceedings of the 8th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (Washington, DC, September 1993), pages 288–303. doi: `10.1145/165854.165907` [acceptance rate 26/278=9%, cited by 45]

54. Hosking AL, Brown E, Moss JEB. Update logging for persistent programming languages: A comparative performance evaluation. *Proceedings of the 19th International Conference on Very Large Data Bases (VLDB)* (Dublin, Ireland, August 1993), pages 429–440, Morgan Kaufmann, 1993. `http://www.vldb.org/conf/1993/P429.PDF` [acceptance rate 54/311=17%, cited by 22]

1992 55. Hosking AL, Moss JEB, Stefanović D. A comparative performance evaluation of write barrier implementations. *Proceedings of the 7th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (Vancouver, Canada, October 1992), pages 92–109, October, 1992. doi: `10.1145/141936.141946` [acceptance rate 31/250=12%, cited by 149]

1990 56. Hosking AL, Moss JEB. Towards compile-time optimisations for persistence. *Proceedings of the 4th International Workshop on Persistent Object Systems (POS)*

(Martha's Vineyard, Massachusetts, September 1990), Dearle, Shaw and Zdonik (Eds), *Implementing Persistent Object Bases: Principles and Practice*, pages 17–27, Morgan Kaufmann, 1990. [acceptance rate 31/82=38%, cited by 30]

Editorials

2001 57. Hosking AL. Session 4: Overview. *Proceedings of the 9th International Workshop on Persistent Object Systems (POS)* (Lillehammer, Norway, September 2000), Kirby (Ed), *Lecture Notes in Computer Science 2135*, pages 157–160, Springer-Verlag, 2001. doi: 10.1007/3-540-45498-5_13

2000 58. Hosking AL, Cutts Q. Persistent object systems. *Software — Practice and Experience 30*(4), pages 293–294, April 2000. http://tinyurl.com/3tqye4x [cited by 2]

Edited volumes

2013 59. Hosking AL, Eugster P, Bolz CF (Eds.). DLS 2013, *Proceedings of the 9th Symposium on Dynamic Languages* (Indianapolis, IN, October 2013). doi: 10.1145/2508168

60. Hosking AL, Eugster P, Lopes CV (Eds.). OOPSLA 2013, *Proceedings of the 2013 ACM SIGPLAN International Conference on Object Oriented Programming, Systems, Languages, and Applications* (Indianapolis, IN, October 2013). doi: 10.1145/2509136

61. Hosking AL, Eugster P (Eds.). SPLASH 2013, *ACM SIGPLAN Conference on Systems, Programming, Languages, and Applications: Software for Humanity* (Indianapolis, IN, October 2013), Companion Volume. doi: 10.1145/2508075

62. Hosking AL, Eugster P, Hirschfeld R (Eds.). Onward! 2013, *Proceedings of the ACM Symposium on New Ideas in Programming and Reflections on Software: Onward!* (Indianapolis, IN, October 2013). doi: 10.1145/2509578

2009 63. Hosking AL, Bacon DF, Krieger O (Eds.). VEE 2009, *Proceedings of the 5th International Conference on Virtual Execution Environments* (Washington, DC, March 2009). doi: 10.1145/1508293

2006 64. Hosking AL, Adl-Tabatabai A-R (Eds.). MSPC 2006, *Proceedings of the 2006 Workshop on Memory System Performance and Correctness* (San Jose, California, October 2006). doi: 10.1145/1178597

2001 65. Chambers C, Hosking AL (Eds.). ISMM 2000, *Proceedings of the 2nd ACM SIGPLAN International Symposium on Memory Management* (Minneapolis, Minnesota, October 2000), *ACM SIGPLAN Notices 36*(1), January, 2001. doi: 10.1145/362422

66. Hosking AL, Cutts Q (Eds.). Special issue on Persistent Object Systems. *Software — Practice and Experience* 30(4), April 2000.

Magazine articles

2004 67. Mueller F, Hosking AL. Penumbra: Simplifying Eclipse. *Dr. Dobb's Journal: Software Tools for the Professional Programmer 365*, pages 62–66, October, 2004. http://www.ddj.com/184405854

Letters

1999 68. Hosking AL. Patriotism alone is not enough. Letter to the Editor commenting on state of academic computer science in Australia, *The Australian — Australia's National Newspaper*, 1st Edition, Wednesday 31st March, 1999, p 38. News Corporation.

Abstracts

1999 69. Hosking AL, Nystrom N, Whitlock D, Cutts Q, Diwan A. Partial redundancy elimination for access path expressions. Abstracted in *Aliasing in Object Oriented Systems*, Noble, Vitek, Lea, Almeida (Eds), *Object-Oriented Technology — ECOOP'99 Workshop Reader*, *Lecture Notes in Computer Science 1743*, pages 136–163, Springer-Verlag, 1999.
doi: 10.1007/3-540-46589-8_8

1997 70. Hosking AL, Novianto AP. Reachability-based orthogonal persistence for C, C++ and other intransigents. *Proceedings of the Workshop on Garbage Collection and Memory Management*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (Atlanta, Georgia, October 1997), 10pp. http://www.dcs.gla.ac.uk/~huw/oopsla97/gc/papers.html [cited by 3]

71. Hosking AL, Cutts Q. Analysing, Profiling and Optimising for Persistence. *Proceedings of the 4th Australian Workshop on Integrated Data Environments* (Magnetic Island, Queensland, Australia, May 1997), 4pp. http://www.cs.adelaide.edu.au/users/idea/idea4

1995 72. Hosking AL. Benchmarking persistent programming languages: quantifying the language/database interface. *Proceedings of the Workshop on Object Database Behavior, Benchmarks, and Performance*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (Austin, Texas, October 1995), 7pp. Abstracted in *Object database behavior, benchmarks, and performance: workshop addendum*, Zorn and Chaudri (Eds), *Addendum to Proceedings of ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)*, *ACM OOPS Messenger 6*(4), pages 159–163, October, 1995.
doi: 10.1145/260111.260272 [cited by 7]

1993 73. Hosking AL, Hudson RS. Remembered sets can also play cards. *Proceedings of the Workshop on Garbage Collection and Memory Management*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (Washington, DC, September 1993), 8pp. ftp://ftp.cs.utexas.edu/pub/garbage/GC93/hosking.ps [cited by 60]

1992 74. Hosking AL. Main memory management for persistence. *Proceedings of the Workshop on Garbage Collection and Memory Management*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (Phoenix, Arizona, October 1991), 6pp. Abstracted in *Garbage collection in object oriented systems*, Wilson and Hayes (Eds), *Addendum to Proceedings of ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)*, *ACM OOPS Messenger 3*(4), pages 63–71, October, 1992.
doi: 10.1145/143776.143792 [cited by 8]

Posters

2015 75. Gammie P, Hosking AL, Engelhardt K. Relaxing Safely: Verification of on-the-fly garbage collection for x86-TSO. *ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)* (Portland, Oregon, June 2015).

2004 76. Mueller F, Hosking AL. Penumbra: An Eclipse plugin for introductory programming. *18th European Conference on Object-Oriented Programming (ECOOP)* (Oslo, Norway, June 2004).

1999 77. Chen J, Hosking AL. Composing Threads with Transactions for Orthogonal Persistence. Poster presented at *14th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (Denver, Colorado, November 1999).

Technical reports

2006 78. Blackburn S, Garner R, Hoffmann C, Khan AM, McKinley KS, Bentzur R, Diwan A, Feinberg D, Frampton D, Guyer S, Hirzel M, Hosking AL, Jump M, Lee H, Moss JEB, Phansalkar A, Stefanović D, VanDrunen T, von Dincklage D, Wiedermann B. The DaCapo benchmarks: Java benchmarking development and analysis (extended version), 30pp. `http://dacapobench.org/dacapo-TR-CS-06-01.pdf` [cited by 20]

2003 79. VanDrunen T, Hosking AL. Corner cases in value-based partial redundancy elimination. Computer Science Technical Report 03-032, 6pp, Purdue University, October, 2003. [cited by 2]

80. Flack C, Hosking A, Vitek J. Idioms in OVM. Computer Science Technical Report 03-017, 8pp, Purdue University, May, 2003. [cited by 14]

1998 81. Nystrom N, Hosking AL, Whitlock D, Cutts Q, Diwan A. Partial redundancy elimination for access path expressions. Computer Science Technical Report 98-044, 12pp, Purdue University, October, 1998.

1996 82. Hosking AL. Residency check elimination for object-oriented persistent languages. Computer Science Technical Report 96-053, 10pp, Purdue University, September, 1996.

1995 83. Hosking AL, Moss JEB. Lightweight write detection and checkpointing for fine-grained persistence. Computer Science Technical Report 95-084, 29pp, Purdue University, November 1995. [cited by 6]

84. Hosking AL. Lightweight support for fine-grained persistence on stock hardware. PhD thesis. Computer Science Technical Report 95-02, 147pp, University of Massachusetts at Amherst, February, 1995. [cited by 17]

1994 85. Moss JEB, Hosking AL, Brown E. Mneme V3.x User's Guide. OSL+SAA Memo 1994-01-V1 (Version 1: May 4, 1994), University of Massachusetts, Amherst, May, 1994

1991 86. Hosking AL, Moss JEB. Compiler support for persistent programming. Computer Science Technical Report 91-25, University of Massachusetts, Amherst, March, 1991 [cited by 17]

1990 87. Hosking AL, Moss JEB, Bliss C. Design of an object faulting persistent Smalltalk. Computer Science Technical Report 90-45, University of Massachusetts, Amherst, May, 1990 [cited by 13]

88. Hosking AL, Moss JEB. Managing persistent data with Mneme: User's guide to the client interface. Object Systems Laboratory, Computer Science Department,

University of Massachusetts, Amherst, March, 1990 [cited by 7]

Refereed Archived Formal Proofs

89. Gammie P, Hosking AL, Engelhardt K. Relaxing Safely: Verified On-the-Fly Garbage Collection for x86-TSO. *Archive of Formal Proofs* (April 2015) `http://afp.sourceforge.net/entries/ConcurrentGC.shtml`

Software

90. *Micro virtual machine:* In collaboration with Professors Steve Blackburn (Australian National University) and Eliot Moss (UMass), and with approximately 1.5 million dollars in funding from both the US National Science Foundation and the Australian Research Council, we are designing a new low-level virtual machine foundation layer for managed language implementations. A μVM captures the insight that there exists a well-defined foundation common to most modern languages that can take responsibility for fundamental abstractions over hardware, concurrency, and memory. By isolating and exposing this substrate, a μVM embodies state-of-the-art base technology available to language implementers while isolating them from the pernicious complexities of these abstractions, freeing them to focus on all-important language-specific optimizations.
91. *Dalvik profiling and benchmarking:* With support from Qualcomm, my student Ahmed Hussein is developing extensive profiling and benchmarking infrastructure for understanding the effects of memory management on power and performance of Dalvik applications.
92. *Software transactional memory for Java:* Compiler and run-time support for open and closed nested transaction support in Java, targeting standard Java virtual machines.
93. *Jikes RVM:* My students and I have been regular contributors to the Jikes RVM research virtual machine for Java (`http://jikesrvm.org`). I participated as a mentor in the Google Summer of Code programme, and oversaw Fil Pizlo's work on state-of-the-art native threading (to replace the old M:N threading implementation) which has since been incorporated into the main release of Jikes RVM. This change to Jikes RVM saw the biggest performance improvement of any previous revision, resulting in a 5–10% improvement for a system that runs close to the speed of commercial JVMs such as HotSpot.
94. *Modula-3 support*: Member of the core team for maintenance of the CM3 and PM3 implementations of the Modula-3 programming languages; responsible for the SPARC/Solaris port. Ported generational/incremental GC support for CM3/PM3 to Mac OS X. Re-wrote GC and threads package in support of native threads on multi-processors, with compiler support for GC barriers. Available from `http://www.modula3.org`
95. *MiniJava*: Implementation of a subset of the Java language targeting native PowerPC (for Linux and Mac OS X) and MIPS (for the SPIM simulator) assembly languages. The subset supports classes and inheritance, static and dynamic fields and methods, native methods, and linking to native libraries. Used as the primary teaching vehicle in both graduate and undergraduate compiler courses at Purdue. The compiler has a significantly revised intermediate stage that is amenable to teaching of advanced compiler techniques such as data-flow

analysis and optimizing transformations. Available on request.

96. *Penumbra*: An Eclipse Plugin for Introductory Programming. A stripped-down perspective for the Eclipse Java development environment, with extensions to improve understanding of Java programs by novice programmers. Penumbra was developed for use by Purdue CS majors in first courses on programming (CS180, CS178). Project funded by IBM. Available from `http://sourceforge.net/projects/purdue-penumbra`
97. *GVNPRE: Global Value Numbering Partial Redundancy Elimination*, by VanDrunen T, Hosking AL. Implemented for the Jikes Research Virtual Machine for Java. Reimplemented by the gcc developers as the recently added GVNPRE-based tree optimization phase in gcc (`http://gcc.gnu.org/wiki/GVN-PRE`).
98. *Persistent Modula-3*: Persistent extension of the Modula-3 programming language. Initial implementation efforts have focused on extending the compiler and run-time system of Digital's Systems Research Center implementation of Modula-3 to support accurate garbage collection and persistence. The prototype is now being used as an experimental platform for comparison of accurate versus conservative garbage collection approaches, and to explore the performance of compiled persistent programming languages. Project funded by Sun Microsystems Laboratories and the National Science Foundation.
    Available from `ftp://ftp.cs.purdue.edu/pub/hosking/pm3`.
99. *BLOAT: Bytecode-Level Optimization and Analysis Tool for Java*, by Nystrom N, Whitlock D, Hosking AL. Permits optimization of Java programs in the absence of source code. In addition to generic optimizations, targeted applications include optimization of persistent extensions to Java. Project funded by Sun Microsystems Laboratories.
    Publicly developed and maintained at `htpp://sourceforge.net/projects/javabloat`. BLOAT is a contributed component of the Polyglot extensible compiler framework (`http://www.cs.cornell.edu/projects/polyglot`), and the db4o object database (`http://www.db4o.com`).
100. *Mneme*: Assisted in the design and implementation of the Mneme persistent object store, intended for use as a low-level object storage manager for advanced database applications. The system is in use at several research sites around the world, and is also being distributed commercially as part of the INQUERY full-text information-retrieval system from the Center for Intelligent Information Retrieval at the University of Massachusetts at Amherst. Users of INQUERY include the Library of Congress, the White House, InfoSeek and the Department of Commerce.
101. *Persistent Smalltalk*: The prototype for much of my thesis research into run-time issues for persistent programming languages. 23,000 lines of C code and 48,000 lines of Smalltalk code implement a full-blown persistent Smalltalk system, including browsers, debuggers, and inspectors, and supporting both garbage collection and lightweight processes (threads).
102. *Language-Independent Garbage Collector Toolkit*: A portable run-time library that supports the implementation of garbage-collectors for arbitrary client programming languages. Several languages have been integrated with the toolkit, notably Smalltalk, Modula-3, SML of New Jersey (CMU and UPenn), and Cecil (Wash-

ington).

Items 98 and 99 are licensed through the Purdue Research Foundation. Items 100, 101 and 102 are distributed and licensed through the Applied Computing Systems Institute of Massachusetts, the University of Massachusetts at Amherst's Computer Science technology-transfer partner.

Colloquia

Invited talks

1. *Power & Performance + Profiling = Pain: Evaluating memory management for mobile*.
   Third NICTA Software Systems Summer School, Sydney, Australia, February 2015.
2. *On verifying concurrent garbage collection for x86-TSO*
   Dagstuhl Seminar 15021, "Concurrent computing in the many-core era", Schloss Dagstuhl — Leibniz-Zentrum für Informatik, January 4–9, 2015.
3. *Modern language abstractions and their run-time systems . . . cracking the nut*.
   University of New South Wales, Sydney, Australia, December 2014.
4. *Verified Concurrent Garbage Collection on the x86-TSO memory model*.
   JTE ASF Managed Runtime, hosted by INRIA/LIP6 in conjunction with Association SIGOPS de France, JTE on topics of managed runtime systems, Laboratoire d'Informatique de Paris 6, June 6 2014.
5. *Towards verified run-time systems: Verifying realistic concurrent garbage collection*.
   Second NICTA Software Systems Summer School, Sydney, Australia, February 2014.
6. *Secret plans and clever tricks: what your VM is not telling you about GC*.
   Seventh Workshop on Virtual Machines and Intermediate Languages (VMIL 2013), SPLASH Workshops, Indianapolis, October 2013.
7. *Open and shut: The case for nested transactions*.
   First NICTA Software Systems Summer School, Sydney, Australia, February 2013.
8. *Schism: a concurrent garbage collector with predictable behavior*.
   National ICT Australia (NICTA) and University of New South Wales, Sydney, Australia, July 2012.
9. *Fragmentation-Tolerant Real-Time Garbage Collection*.
   Keynote, Australian Workshop on Programming Languages and Operating Systems (AusPLOS), National ICT Australia (NICTA) and University of New South Wales, Sydney, Australia, January 2012.
10. *X10 on the Single-Chip Cloud Computer*.
    Many-Core Applications Research Community Symposium, Intel Corporation, Santa Clara, California, March 2011.
11. *Schism: fragmentation-tolerant real-time garbage collection*.
    Australian National University, Canberra, Australia, July 2010.
12. *Schism: fragmentation-tolerant real-time garbage collection*.
    University of St Andrews, Scotland, May 2010.
13. *Schism: fragmentation-tolerant real-time garbage collection*.
    Microsoft Research UK, Cambridge, England, May 2010.
14. *Pervasive virtualisation of Java classes with application to transparent distribution*.

Australian National University, Canberra, Australia, April 2009.

15. *Language extensions for open nested transactions*.
    University of Kent at Canterbury, England, October 2008.
16. *HTM: hardware transactional memory*.
    University of Kent at Canterbury, England, October 2008.
17. *Language extensions for open nested transactions*.
    Microsoft Research UK, Cambridge, England, October 2008.
18. *Language extensions for open nested transactions*.
    University of St Andrews, Scotland, October 2008.
19. *Language extensions for open nested transactions*.
    University of Glasgow, Scotland, October 2008.
20. *Language extensions for open nested transactions*.
    Australian National University, Canberra, Australia, May 2007.
21. *Language extensions for open nested transactions*.
    University of Adelaide, Australia, May 2007.
22. *Language extensions for open nested transactions*.
    Indiana University-Purdue University Indianapolis, March 2007.
23. *Language extensions for open nested transactions*.
    Microsoft Research, Redmond, Washington, March 2007.
24. *Portable, mostly-concurrent, mostly-copying garbage collection for multi-processors*.
    University of Kent at Canterbury, England, May 2006.
25. *Portable, mostly-concurrent, mostly-copying garbage collection for multi-processors*.
    University of Copenhagen, May 2006.
26. *Revocation techniques for Java concurrency*.
    Workshop on Object Systems and Software Architectures, Victor Harbour, South Australia, January 2006
27. *Transactional, persistent, managed runtime environments*.
    Microsoft Research Rotor RFP Capstone Workshop, Redmond, WA, September 2005
28. *Transactional, persistent, managed runtime environments*.
    Microsoft Research Academic Summit, Sao Paulo, Brazil, May 2005
29. *Transactional monitors for concurrent objects*.
    Harvard University, Cambridge, Massachusetts, August 2004
30. *Software prefetching for mark-sweep garbage collection: hardware analysis & software redesign*.
    University of Kent at Canterbury, England, June 2004
31. *Transactional threads for orthogonal persistence*.
    University of Adelaide, Australia, May 2003
32. *Transactional threads for orthogonal persistence*.
    Australian National University, Canberra, Australia, May 2003
33. *Capitulating loads: alternatives to prefetching for garbage collection*.
    NSF Mid-Project Review of *ITR/SW: Dynamic Cooperative Performance Optimization*, University of Massachusetts at Amherst, April 2003
34. *On the usefulness of accuracy and liveness for garbage collection and leak detection*.
    University of Adelaide, Australia, November 2001
35. *On the usefulness of accuracy and liveness for garbage collection and leak detection*.

University of Glasgow, Scotland, September 2001

36. *Optimizing with Persistence*.
    Dagstuhl Seminar 00451, "Effective Implementation of Object-Oriented Programming Languages", Schloss Dagstuhl — Leibniz-Zentrum für Informatik, November 5–10, 2000.
37. *PM3: An orthogonally persistent systems programming language*.
    University of Adelaide, Australia, July 2000
38. *A framework for persistence-enabled optimization of Java applications*.
    Australian National University, Canberra, July 2000
39. *A framework for persistence-enabled optimization of Java applications*.
    University of Glasgow, Scotland, May 2000
40. *PM3: an orthogonally persistent systems programming language*.
    University of St Andrews, Scotland, September 1999
41. *PM3: an orthogonally persistent systems programming language*.
    Stanford University, California, July 1999
42. *High-performance persistent systems: realizing the dream*.
    Australian National University, Canberra, March 1999
43. *Partial redundancy elimination for access expressions with application to orthogonal persistence*.
    Australian National University, Canberra, July 1998
44. *Reachability-based orthogonal persistence for C, C++ and other intransigents*.
    5th Annual FIDE EC/US Workshop in Persistence, Kinlochrannoch, Scotland, August 1997
45. *Optimizing for persistence*.
    Sun Microsystems Laboratories, Mountain View, California, March 1997
46. *Lightweight support for fine-grained persistence on stock hardware*.
    James Cook University, Townsville, Australia, July 1995
47. *Lightweight support for fine-grained persistence on stock hardware*.
    Sun Microsystems Laboratories (East), Massachusetts, May 1995
48. *Towards high-performance persistent programs*.
    Harvard University, Cambridge, Massachusetts, April 1994
49. *Towards high-performance persistent programs*.
    Yale University, New Haven, Connecticut, April 1994
50. *Towards high-performance persistent programs*.
    University of Maryland, College Park, April 1994
51. *Towards high-performance persistent programs*.
    University of Illinois at Urbana-Champaign, April 1994
52. *Towards high-performance persistent programs*.
    Purdue University, West Lafayette, Indiana, April 1994
53. *Towards high-performance persistent programs*.
    University of Maryland, Baltimore County, March 1994
54. *Towards high-performance persistent programs*.
    York University, Toronto, Canada, March 1994
55. *Towards high-performance persistent programs*.
    University of Illinois at Chicago, March 1994
56. *On trial: virtual memory primitives for user programs*.

Carnegie Mellon University, Pittsburgh, Pennsylvania, June 1993

57. *Implementing efficient persistent programming languages*. James Cook University, Townsville, Australia, March 1992
58. *Implementing efficient persistent programming languages*. University of Sydney, Australia, March 1992

Conferences and workshops

presenter in bold

59. **Lin Y**, Wang K, Blackburn SM, Hosking AL, Norrish M. Stop and Go: Understanding yieldpoint behavior. *ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Portland, Oregon, June 2015).
60. **Hussein A**, Hosking AL, Payer M, Vick C. Don't race the memory bus: Taming the GC leadfoot. *ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Portland, Oregon, June 2015).
61. **Gammie P**, Hosking AL, Engelhardt K. Relaxing Safely: Verification of on-the-fly garbage collection for x86-TSO. *ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)* (Portland, Oregon, June 2015).
62. Hussein A, Payer M, **Hosking AL**, Vick C. Impact of GC Design on Power and Performance for Android. *ACM International Conference on Systems and Storage (SYSTOR)* (Haifa, Israel, May 2015).
63. **Wang K**, Lin Y, Blackburn SM, Norrish M, Hosking AL. Draining the Swamp: Micro virtual machines as solid foundation for language development. *Inaugural Summit on Advances in Programming Languages (SNAPL)* (Asilomar, California, May 2015).
64. Chapman K, **Hosking AL**, Moss JEB, Richards T. Closed and Open Nested Atomic Actions for Java: Language Design and Prototype Implementation. *International Conference on the Principles and Practice of Programming on the Java platform: virtual machines, languages, and tools (PPPJ)* (Kraków, Poland, September 2014).
65. **Yang X**, Blackburn SM, Frampton D, Hosking AL. Barriers reconsidered, friendlier still! *ismm* (Beijing, China, June 2012).
66. **Pizlo F**, Frampton D, Hosking AL. Fine-grained adaptive biased locking. *International Conference on the Principles and Practice of Programming on the Java platform: virtual machines, languages, and tools (PPPJ)* (Kongens Lyngby, Denmark, August 2011).
67. **Chapman K**, Hussein A, Hosking AL. X10 on the Single-chip Cloud Computer. *X10 Workshop (X10)* (San Jose, California, June 2011).
68. **Pizlo F**, Ziarek L, Maj P, Hosking AL, Blanton E, Vitek J. Schism: fragmentation-tolerant real-time garbage collection. *ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)* (Toronto, Canada, June 2010).
69. **Hellyer L**, Jones R, Hosking AL. The locality of concurent write barriers. *ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Toronto, Canada, June 2010).
70. **Kalibera T**, Pizlo F, Hosking AL, Vitek J. Scheduling hard real-time garbage collection. *30th IEEE Real-Time Systems Symposium (RTSS)* (Washington, DC, December 2009).
71. **McGachey P**, Hosking AL, Moss JEB. Classifying Java class transformations for pervasive virtualized access. *8th ACM Conference on Generative Programming and*

Component Engineering (GPCE) (Denver, Colorado, October 2009).

72. **McGachey P**, Hosking AL, Moss JEB. Pervasive load-time transformation for transparently distributed Java. *4th International Workshop on Bytecode Semantics, Verification, Analysis and Transformation (BYTECODE)* (York, England, March 2009).
73. **Hambrusch SE**, Hoffmann C, Korb JT, Haugan M, Hosking AL. A multidisciplinary approach towards computational thinking for science majors. *ACM SIGCSE Technical Symposium on Computer Science Education (SIGCSE)* (Chattanooga, Tennessee, March 2009).
74. **Pizlo F**, Hosking AL, Vitek J. Hierarchical real-time garbage collection. *ACM SIGPLAN/SIGBED conference on Languages, Compilers, and Tools for Embedded Systems (LCTES)* (San Diego, California, June 2007).
75. Ni Y, **Menon V**, Adl-Tabatabai A, Hosking AL, Hudson RL, Moss JEB, Saha B, Shpeisman T. Open nesting in software transactional memory. *ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP)*, (San Jose, California, March 2007).
76. **Blackburn S**, Garner R, Hoffmann C, Khan A, McKinley KS, Bentzur R, Diwan A, Feinberg D, Frampton D, Guyer S, Hirzel M, Hosking A, Jump M, Lee H, Moss JEB, Phansalkar A, Stefanović D, VanDrunen T, von Dincklage D, Wiedermann B. The DaCapo benchmarks: Java benchmarking development and analysis. *21st ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (Portland, Oregon, October 2006).
77. **Welc A**, Hosking AL, Jagannathan S. Transparently reconciling transactions with locking for Java synchronization. *20th European Conference on Object-Oriented Programming (ECOOP)* (Nantes, France, July 2006).
78. **Hosking AL**. Portable, mostly-concurrent, mostly-copying garbage collection for multi-processors. *ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Ottawa, Canada, June 2006).
79. **McGachey P**, Hosking AL. Reducing generational copy reserve overhead with fallback compaction. *ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Ottawa, Canada, June 2006).
80. **Welc A**, Jagannathan S, Hosking AL. Safe futures for Java. *20th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (San Diego, California, October 2005).
81. **Blackburn S**, Hosking AL. Barriers: friend or foe? *ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Vancouver, Canada, October 2004).
82. **Cher C-Y**, Hosking AL, Vijaykumar TN. Software prefetching for mark-sweep garbage collection: hardware analysis & software redesign. *11th ACM Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)* (Boston, Massachusetts, October 2004).
83. **Welc A**, Hosking AL, Jagannathan S. Preemption-based avoidance of priority inversion for Java. *International Conference on Parallel Processing* (Montréal, Canada, August 2004).
84. **Welc A**, Jagannathan S, Hosking AL. Transactional monitors for concurrent objects. *18th European Conference on Object-Oriented Programming (ECOOP)* (Oslo, Norway, June 2004).

85. **Vitek J**, Jagannathan S, Welc A, Hosking AL. A semantic framework for designer transactions. *European Symposium on Programming, European Joint Conferences on Theory and Practice of Software* (Barcelona, Spain, March/April 2004).
86. **VanDrunen T**, Hosking AL. Value-based partial redundancy elimination. *International Conference on Compiler Construction, European Joint Conferences on Theory and Practice of Software* (Barcelona, Spain, March/April 2004).
87. **Mueller F**, Hosking AL. Penumbra: an Eclipse plugin for introductory programming. *Workshop on Eclipse Technology eXchange (eTX)*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA).
88. **Hirzel M**, Diwan A, Hosking AL. On the usefulness of liveness for garbage collection and leak detection. *15th European Conference on Object-Oriented Programming (ECOOP)* (Budapest, Hungary, June 2001).
89. **Whitlock D**, Hosking AL. A framework for persistence-enabled optimization of Java Object Stores. *9th International Workshop on Persistent Object Systems (POS)* (Lillehammer, Norway, September 2000).
90. **Hosking AL**, Chen J. Mostly-copying reachability-based orthogonal persistence. *14th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (Denver, Colorado, November 1999).
91. **Hosking AL**, Chen J. PM3: an orthogonally persistent systems programming language — Design, Implementation, Performance. *25th International Conference on Very Large Data Bases (VLDB)* (Edinburgh, Scotland, September 1999).
92. **Hosking AL**, Nystrom N, Whitlock D, Cutts Q, Diwan A. Partial redundancy elimination for access path expressions. *Intercontinental Workshop on Aliasing in Object-Oriented Systems* (Lisbon, Portugal, June 1999)
93. **Hosking AL**, *Persistent Modula-3* demonstration. *ACM SIGPLAN International Symposium on Memory Management (ISMM)* (Vancouver, Canada, October 1998)
94. Brahnmath K, Nystrom N, **Hosking AL**, Cutts Q. Swizzle barrier optimizations for orthogonal persistence in Java. *3rd International Workshop on Persistence and Java (PJ)* (Tiburon, California, August 1998).
95. **Hosking AL**, Nystrom N, Cutts Q, Brahnmath K. Optimizing the read and write barriers for orthogonal persistence. *8th International Workshop on Persistent Object Systems (POS)* (Tiburon, California, August 1998).
96. **Cutts QI**, Lennon S, Hosking AL. Reconciling buffer management with persistence optimizations. *8th International Workshop on Persistent Object Systems (POS)* (Tiburon, California, August 1998).
97. **Hosking AL**, Novianto AP. Mostly-copying reachability-based orthogonal persistence for C, C++ and other intransigents. *Workshop on Garbage Collection and Memory Management*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (Atlanta, Georgia, October 1997).
98. Cutts Q, **Hosking AL**. Analysing, Profiling and Optimising Orthogonal Persistence for Java. *2nd International Workshop on Persistence and Java (PJ)* (Half Moon Bay, California, August 1997).
99. *Analysing, Profiling and Optimising for Persistence*. 4th Australian Workshop on Integrated Data Environments, Magnetic Island, Queensland, Australia, May

1997

100. Moss JEB, **Hosking AL**. Approaches to adding persistence to Java. *1st International Workshop on Persistence and Java (PJ)* (Drymen, Scotland, September 1996).
101. **Hosking AL**. Residency check elimination for object-oriented languages. *7th International Workshop on Persistent Object Systems (POS)* (Cape May, New Jersey, May 1996).
102. **Moss JEB**, Hosking AL. Expressing object residency optimizations using pointer type annotations. *6th International Workshop on Persistent Object Systems (POS)* (Tarascon, France, September 1994).
103. **Hosking AL**. Benchmarking persistent programming languages: quantifying the language/database interface. *Workshop on Object Database Behavior, Benchmarks, and Performance*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (Austin, Texas, October 1995).
104. **Hosking AL**, Moss JEB. Protection traps and alternatives for memory management of an object-oriented language. *14th Symposium on Operating Systems Principles* (Asheville, NC, December 1993).
105. **Hosking AL**, Moss JEB. Object fault handling for persistent programming languages: A performance evaluation. *8th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (Washington, DC, September 1993).
106. **Hosking AL**, Hudson RS. Remembered sets can also play cards. *Workshop on Garbage Collection and Memory Management*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (Washington, DC, September 1993).
107. **Hosking AL**, Brown E, Moss JEB. Update logging for persistent programming languages: A comparative performance evaluation. *19th International Conference on Very Large Data Bases (VLDB)* (Dublin, Ireland, August 1993).
108. **Hosking AL**, Moss JEB, Stefanović D. A comparative performance evaluation of write barrier implementations. *7th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (Vancouver, Canada, October 1992).
109. **Hosking AL**. Main memory management for persistence. *Workshop on Garbage Collection and Memory Management*, ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) (Phoenix, Arizona, October 1991).
110. **Hosking AL**, Moss JEB. Towards compile-time optimisations for persistence. *4th International Workshop on Persistent Object Systems (POS)* (Martha's Vineyard, Massachusetts, September 1990).

Professional service

ACM SIGPLAN Programming Languages Software Award committee

2014 Member

ACM SIGPLAN executive committee

2009-2012 Member-at-large

Conference general chair

2015 ACM SIGPLAN International Symposium on Memory Management (ISMM)

2013 ACM SIGPLAN Conference on Systems, Programming, Languages and Applications: Software for Humanity (SPLASH), featuring OOPSLA Research Papers, Onward!, Dynamic Languages Symposium (DLS), etc.

2012 26th European Conference on Object-Oriented Programming (ECOOP), Beijing, China

2009 ACM SIGPLAN/SIGOPS Conference on Virtual Execution Environments (VEE)

2006 ACM SIGPLAN Workshop on Memory Systems Performance and Correctness (MSPC)

Conference programme chair

2013 Third Workshop on Runtime Environments, Systems, Layering and Virtualized Environments (RESoLVE), colocated with ASPLOS'13

2000 ACM SIGPLAN International Symposium on Memory Management (ISMM)

Conference sponsorship chair

2015 ACM SIGPLAN Conference on Systems, Programming, Languages and Applications: Software for Humanity (SPLASH), featuring OOPSLA Research Papers, Onward!, Dynamic Languages Symposium (DLS), etc.

Peer reviewer

2012 Excellence in Research for Australia (ERA). ERA assesses research quality within Australia's 41 higher education providers using a combination of indicators and expert review by committees comprising experienced, internationally-recognised experts.

Guest editor

2000 Special issue on *Persistent Object Systems*, *Software — Practice and Experience 30*, 4 (April 2000)

Editorial panel member

2006 *Science of Computer Programming (SCP)*, special issue on *Memory Management*

2005 *Science of Computer Programming (SCP)*, special issue on *Synchronization and Concurrency in Object-Oriented Languages*

Conference programme committee member

2015 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI) external revieew committee

2014 ACM SIGPLAN International Symposium on Memory Management (ISMM) external review committee

2014 ACM SIGPLAN/SIGOPS Conference on Virtual Execution Environments (VEE) programme committee

2014 ACM Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS) external review committee

2013 ACM SIGPLAN International Symposium on Memory Management (ISMM) external review committee

2012 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI) programme committee

2011 5th Workshop on Virtual Machines and Intermediate Languages, programme committee

2011 ACM SIGPLAN Conference on Languages, Compilers, and Tools for Embedded Systems (LCTES) programme committee

2010 ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) programme committee

2010 ACM SIGPLAN International Symposium on Memory Management (ISMM) external review committee

2010 ACM SIGPLAN/SIGOPS Conference on Virtual Execution Environments (VEE) programme committee

2009 International Conference on Object Oriented Data Bases (ICOODB) programme committee

2009 ACM SIGPLAN International Symposium on Memory Management (ISMM) external review committee

2009 ACM SIGPLAN/SIGACT Symposium on Principles of Programming Languages (POPL) programme committee

2008 ACM Conference on Principles and Practice of Programming in Java (PPPJ) programme committee

2008 ACM SIGPLAN International Symposium on Memory Management (ISMM) expert review committee

2007 ACM SIGPLAN International Symposium on Memory Management (ISMM) programme committee

2007 ACM SIGPLAN/SIGOPS Conference on Virtual Execution Environments (VEE) programme committee

2005 Workshop on Synchronization and Concurrency in Object-Oriented Languages (SCOOL) programme committee

2005 European Conference on Object-Oriented Programming (ECOOP) programme committee

2004 Workshop on Eclipse Technology eXchange (eTX) programme committee

2004 ACM SIGPLAN Symposium on Memory System Performance (MSP) programme committee

2002 ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) programme committee

2001 International Conference on Very Large Data Bases (VLDB) programme committee

2000 International Workshop on Persistent Object Systems (POS) programme committee

2000 ACM SIGPLAN International Symposium on Memory Management (ISMM) programme committee

2000 International Symposium on Objects and Databases (SODB) programme committee

1998 ACM SIGPLAN International Symposium on Memory Management (ISMM) programme committee

1998 ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) programme committee

1997 International Workshop on Persistence and Java (PJ) programme committee

1996 International Workshop on Persistent Object Systems (POS) programme committee

Other conference refereeing

2008 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)

2007 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)

2006 ACM SIGPLAN International Conference on Functional Programming (ICFP)

2003 ACM SIGPLAN Conference on Languages, Compilers, and Tools for Embedded Systems (LCTES)

2002 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)

2001 ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)

2000 ACM SIGPLAN International Conference on Functional Programming (ICFP)

2000 ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)

2000 European Conference on Object-Oriented Programming (ECOOP)

2000 Symposium on Objects and Databases

1998 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)

1997 ACM International Conference on Supercomputing (SC)

1997 ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)

1996 USENIX Symposium on Operating Systems Design and Implementation (OSDI)

1996 International Conference on High-Performance Computing (HPC)

1996 ACM International Symposium on Symbolic and Algebraic Computation (ISSAC)

1994 USENIX Symposium on Operating Systems Design and Implementation (OSDI)

1992 ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)

1991 ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)

1991 International Conference on Very Large Data Bases (VLDB)

1991 International Workshop on Database Programming Languages (DBPL)

Conference steering committee chair

2012–2013 ACM SIGPLAN/SIGOPS Conference on Virtual Execution Environments (VEE)

Conference steering committee member

2015– ACM SIGPLAN International Symposium on Memory Management (ISMM)

2013– ACM SIGPLAN Conference on Systems, Programming, Languages and Applications: Software for Humanity (SPLASH), featuring OOPSLA Research Papers, Onward!, Dynamic Languages Symposium (DLS), etc.

2013– ACM SIGPLAN Symposium on New Ideas in Programming and Reflections on Software (Onward!)

2009–2013 ACM SIGPLAN/SIGOPS Conference on Virtual Execution Environments (VEE)

2006–2013 ACM Transact: Workshop on Languages, Compilers, and Hardware Support for Transactional Computing (TRANSACT)

2000–2006 ACM SIGPLAN International Symposium on Memory Management (ISMM)

Journal refereeing

- ACM Transactions on Architecture and Code Optimization (TACO)
- Software — Practice and Experience (SPE)
- ACM Transactions on Embedded Computing Systems (TECS)
- ACM Transactions on Programming Languages and Systems (TOPLAS)
- ACM Computing Surveys
- The International Journal on Very Large Data Bases (VLDBJ)
- Science of Computer Programming (SCP), special issue on synchronization and concurrency in object-oriented languages
- Science of Computer Programming (SCP), special issue on memory management
- Communications of the ACM (CACM)
- International Journal on Theory and Practice of Object Systems (TAPOS)

Research proposal refereeing

2014 National Science Foundation (NSF)

2014 Israel Science Foundation

2011 National Science Foundation (NSF)

2011 Qatar National Research Fund (QNRF)

2010 National Science Foundation (NSF)

2010 Austrian Science Fund

2010 Israel Science Foundation

2009 Royal Society of New Zealand, Marsden Fund

2007 National Science Foundation (NSF)

2003 National Science Foundation (NSF)

2003 Royal Society of New Zealand, Marsden Fund

2003 Natural Sciences and Engineering Research Council of Canada (NSERC)

2002 Natural Sciences and Engineering Research Council of Canada (NSERC)

2001 Royal Society of New Zealand, Marsden Fund

External examiner/examining committee member

2014 Kunshan Wang, PhD, Australian National University, *Concurrent Execution in Managed Languages*

2014 Yi Lin, PhD, Australian National University, *An Efficient Implementation of μVM: A Substrate Virtual Machine For Managed Languages*

2012 Yi Lin, MPhil, Australian National University, *Virtual Machine Design and High-Level Implementation Languages*

2010 Daniel Frampton, PhD, Australian National University, *Garbage Collection and the Case for High-level Low-level Programming*, winner of the 2010 Computing Research and Education Australian Distinguished Doctoral Dissertation

2009 Ali Ibrahim, PhD, University of Texas at Austin, *Practical Transparent Persistence*

2006 Niels Elgaard Larsen, University of Copenhagen, *Emerald Database — Integrating Transactions, Queries, and Method Indexing into a system based on mobile objects*

2005 John Zigman, PhD, Australian National University, *A General Framework for the Description and Construction of Hierarchical Garbage Collection Algorithms*

2003 Zhen He, PhD, Australian National University, *Integrated Buffer Management for Object-Oriented Database Systems*

2002 Stuart Blair, PhD, University of Glasgow, Scotland, *On the Classification and Evaluation of Prefetching Schemes*

2000 Antonios Printezis, PhD, University of Glasgow, Scotland, *Management of Long-Running High-Performance Persistent Object Stores*

1999 M Schulz, MSc, University of Cape Town, *Garbage Collection of the PJava Object Store*

Mentoring

2010 Google Summer of Code, Jikes RVM. Graph coloring register allocation

2008 Google Summer of Code, Jikes RVM. Guided Fil Pizlo's work on retargeting the Jikes Research Virtual Machine to native threads (pthreads), replacing the old user-level threads implementation. This was merged into the mainline code base of Jikes RVM in 2009.

Research funding

Awards

$286,300 Vijaykumar TN, Kulkarni M, Pai V, Hosking AL, *II-New: A Cluster of Nodes with 32 Cores and 256-GB Memory to Enable Many-Core Systems Research and Education*, National Science Foundation (CNS-1405939), 8/1/2014–7/31/2017

$1,200,000 Hosking AL, Moss JEB (UMass), Richards T (UMass), *SHF: Medium: Collaborative: Micro Virtual Machines for Managed Languages — Abstraction, defined and contained*, National Science Foundation (CCF-1408896: $642,319; CCF-1409284: $557,681), 6/15/2014–6/14/2018.

$50,000 Hosking AL, *Dalvik Garbage Collection — Memory Management for Mobile*, Qualcomm, 2014.

A$300,000 Blackburn SM (Australian National University), Hosking AL, *Micro Virtual Machines: Abstraction, contained*, Australian Research Council Discovery Grant (DP140103878), 2014-2016.

$50,000 Hosking AL, *Dalvik Garbage Collection — Memory Management for Mobile*, Qualcomm, 2013.

$12,000 Hosking AL, *SPLASH 2013 Travel Support*, National Science Foundation (CCF-1347630), 9/1/2013–8/31/2014.

$100,000 Hosking AL, *Dalvik Garbage Collection — Memory Management for Mobile*, Qualcomm, 2013.

$338,030 Hosking AL, *CSR: Medium: Collaborative: Portable Performance for Parallel Managed Languages Across the Many-Core Spectrum*, National Science Foundation (CNS-1161237), 5/15/2012–5/14/2015.

$46,800 Vitek J, Hosking AL, Krintz C (UCSB), Padua D (U Illinois), *VEESC: Virtual Execution Environments for Scientific Computing*, National Science Foundation (CI-1042905), 9/1/2010–8/31/2011.

$30,000 Hosking AL, *X10 Innovation Award: Transactions for reliable distributed X10 execution*, IBM, 2010.

$219,000 Hosking AL, Jagannathan S, Vitek J, Grama A. *Language and Runtime Support for Safe and Scalable Programs*, Microsoft, 2009.

$39,000 Hosking AL, *Delivering on Atomic Actions: Unlocking Concurrency for Ordinary Programmers*, Intel, 2009.

$406,000 Vitek J, Hosking AL, *Certified Garbage Collection for Highly Responsive Systems*, Computing Processes and Artifacts Program, Division of Computer and Communicaion Foundations, National Science Foundation (CCF-0811691), 8/1/2008–7/31/2011. [acceptance rate 10-15% of 514 submitted]

$109,500 Hosking AL, Jagannathan S, Vitek J, Grama A, *Language and Runtime Support for Safe and Scalable Programs*, Microsoft, 2008.

£26,456 Jones RE (University of Kent at Canterbury), Hosking AL, *Visiting Researcher: Dr. Tony Hosking*, Engineering and Physical Sciences Research Council, United Kingdom (EP/F06523X/1), 1 May 2008 – 31 October 2008.

$40,000 Hosking AL, *Delivering on Atomic Actions: Unlocking Concurrency for Ordinary Programmers*, Intel, 2008.

$459,000 Hambrusch S, Hosking A, Kais S, Haugan M, Hoffmann C, *CPATH CB: Computing Education in Science Context*, CISE Pathways to Integrated Undergraduate Computing Education (CPATH) Program, Division of Computer and Communication Foundations, National Science Foundation (CCF-0722210), 7/1/2007–6/30/2009. [acceptance rate 25% of 141 submitted]

$30,000 Hosking AL, *CSR-AES Collaborative: Encore/J: Transparently Recoverable Java for Resilient Distributed Computing*, Computer Systems Research, Division of Computer and Network Systems, National Science Foundation (CNS-0720505), 9/1/2007–8/31/2008. [acceptance rate 10-15% of 410 submitted]

$275,001 Hosking AL, *CPA: Scalable Concurrent Compacting Garbage Collection for Commodity Multi-Core Processors*, Computing Processes and Artifacts, Division of Computer and Communication Foundations, National Science Foundation (CPA-0702240), 7/15/2007–7/14/2010. [acceptance rate 12% of 525 submitted]

$40,000 Hosking AL, *Delivering on Atomic Actions: Unlocking Concurrency for Ordinary Programmers*, Intel, 2007.

$25,000 Hosking AL, *Open nesting abstractions with Bartok and Singularity*, Microsoft, 2007.

$6,000 Hosking AL, *ST-CRTS: Collaborative: Delivering on Atomic Actions: Unlocking Concurrency for Ordinary Programmers*, Research Education for Undergraduates (REU) supplement, Computing Processes and Artifacts, Division of Computer and Communication Foundations, National Science Foundation (CCF-0733139), 2/1/2006–1/31/2009.

$2,000 Hosking AL, *Multi-core University Research*, Intel, 2005.

$99,979 Jagannathan S, Vitek J, Grama A, Hosking AL, *CRI: A Computational Infrastructure for Experimentation on Relaxed Concurrency Abstractions and their Applications*, Computing Research Infrastructure Program, Division of Computer and Network Systems, National Science Foundation (CNS-0551658), 3/1/2006–2/29/2008. [acceptance rate 12-15% of 229 submitted]

$279,999 Hosking AL, *ST-CRTS: Collaborative: Delivering on Atomic Actions: Unlocking Concurrency for Ordinary Programmers*, Computing Processes and Artifacts Program, Division of Computer and Communication Foundations, National Science Foundation (CCF-0540866), 2/1/2006–1/31/2009. [acceptance rate 10% of 532 submitted]

$30,000 Hosking AL, *CSR-AES: Collaborative: RuggedJ: Resilient Distributed Java Over Heterogeneous Platforms*, Computer Systems Research Program, Division of Computer and Network Systems, National Science Foundation (CNS-0509377), 9/1/2005–8/31/2006. [acceptance rate 10-15% of 440 submitted]

$27,000 Hosking AL, *Eclipse Innovation Award*, IBM, 2005

$32,000 Hosking AL, *Transactional, Persistent, Managed Runtime Environments*, Microsoft, 2004

$25,000 Hosking AL, *Eclipse Innovation Award: An Eclipse plugin for introductory programming*, IBM, 2004

$30,000 Palsberg J, Hosking AL, *Eclipse Innovation Award: Teaching introductory computer science using Eclipse*, IBM, 2003

$30,000 Vitek J, Hosking AL, *ReAssure — resilient and secure virtual machines for network computing*, Lilly Endowment for support of the Center for Education and Research in Information Assurance and Security, Purdue University, 2001

$3,081,601 Vitek J, Hosking AL, Palsberg J (UCLA), Pugh W (U Maryland), Lea D (SUNY Oswego), *DCMF/NES: Dynamic compositional middleware frameworks for networked embedded systems*, PCES Program, DARPA/ITO via Air Force Research Laboratory, June 2001–August 2004

$3,203,141 Moss JEB (UMass), McKinley KS (U Texas), Weems CC (UMass), Hosking AL , Diwan A (Colorado), Stefanović (New Mexico), *ITR/SW: Dynamic cooperative performance optimization*, Information Technology Research Program, National Science Foundation (CCR-0085792), 9/1/2000–8/31/2005.

$215,000 Hosking AL, *Transactional threads for reliable persistent application systems*, Information and Data Management Program, Division of Information and Intelligent Systems, Directorate for Computer and Information Science and Engineering, National Science Foundation (IIS-9988637), 9/1/2000–8/31/2003.

$41,558 Vitek J, Hosking AL, *ReAssure — resilient and secure virtual machines for network computing*, Lilly Endowment for support of the Center for Education and Research in Information Assurance and Security, Purdue University, 2000

$40,000 Hosking AL, *Analyzing, profiling and optimizing orthogonal persistence for Java*, Sun Microsystems Laboratories Collaborative Research Grant, 1999

$35,000 Hosking AL, *Analyzing, profiling and optimizing Java applications*, IBM Shared University Research, 1998

$40,000 Hosking AL, *Analyzing, profiling and optimizing orthogonal persistence for Java*, Sun Microsystems Laboratories Collaborative Research Grant, 1998

$10,500 Comer D, Hosking AL, *Crosspoint extension*, Intel Corporation, 1998

$64,000 Comer D, Hosking AL, *Crosspoint extension*, Intel Corporation, 1998

$185,000 Hosking AL, *Compiling with persistence*, Operating Systems and Compilers Program, Division of Computer-Communications Research, Directorate for Computer and Information Science and Engineering, National Science Foundation (CCR-9711673), 8/1/1997–7/31/2000.

$75,000 Hosking AL, *Analyzing, profiling and optimizing orthogonal persistence for Java*, Sun Microsystems Laboratories Collaborative Research Grant, 1997

$5,000 Hosking AL, *Compiling for persistence*, Purdue Research Foundation Summer Faculty Grant, 1997

$2,500 Hosking AL, *High-performance persistent information systems*, Purdue Global Initiative Grant, International Programs, 1996

$70,000 Hosking AL, *Compiler support for persistence*, Sun Microsystems Laboratories Collaborative Research Grant, 1996

$7,000 Hosking AL, *Language and compiler support for fine-grained object migration*, Purdue University Department of Computer Science Seed Fund Grant, 1995

$5,000 Hosking AL, *Compiler support for garbage collection and persistence*, Purdue University Department of Computer Science Seed Fund Grant, 1995

Donations (list price)

$4,050 Qualcomm Snapdragon 8x60 Mobile Development Platform with Android, 2012

$25,000 Sun Microsystems equipment, 2007

$5,000 Microsoft software, 1997

$35,925 Sun Microsystems equipment, 1997

$5,000 Microsoft software, 1996

Graduate students supervised

PhD committee chair

expected 2012 **Filip Pizlo**, to Software Engineer, Apple Computer, Cupertino, California.
Fragmentation tolerant real time garbage collection

May 2010 **Phil McGachey**, to Senior Engineer, VMWare, Boston, Massachusetts.
Transparent distribution for Java applications

May 2006 **Adam Welc**, to Research Scientist, Intel, Santa Clara, California.
Concurrency abstractions for programming languages using optimistic protocols

August 2004 **Thomas VanDrunen**, to Assistant Professor, Wheaton College, Wheaton, Illinois.
Partial redundancy elimination for global value numbering

December 2004 **Krzysztof Palacz**, to Sun Microsystems Laboratories, Mountain View, California.
Crusoe — Towards a multicomputer execution environment for Java

Master's thesis committee chair

December 2013 **Ahmed Hussein**.
On tracing the memory behavior of Dalvik applications

December 2005 **Phil McGachey**.
An improved generational copying garbage collector

May 2004 **Frank Mueller**.
Penumbra: An Eclipse plugin for introductory programming

May 2000 **David Whitlock**, to GemStone Systems and Adjunct Faculty Member at Portland State University, Portland, Oregon.
Persistence-enabled optimization of Java programs

August 1998 **Nathaniel Nystrom**, to Cornell University PhD program via Hewlett-Packard.
Bytecode-level analysis and optimization of Java classes

May 1998 **Kumar Brahnmath**, to American Management Systems.
Optimizing Orthogonal Persistence for Java

May 1996 **Nedim Fresko**, to Sun Microsystems, Palo Alto, California.
Compiler support for accurate garbage collection on RISC processors

Other funded research assistants

2011– **Qi Chen**.

2009– **Ahmed Abd-ElHaffiez Hussein**.

2009– **Keith Chapman**.

December 2008 **Athul Acharya**, to Intel.

May 2004 **Chen-Yong Cher**, *PhD in Electrical and Computer Engineering*, to IBM Research, Yorktown Heights.
Exploring and Evaluating Control-Flow and Thread-Level Parallelism

August 2003 **Kailash Agrawal**, to Advanced Micro Devices, Austin, Texas.

2002 **Antonio Ake**.

May 2000 **Jiawan Chen**.

May 2000 **Aria Novianto**.

Other PhD committee memberships

Purdue CS unless otherwise noted

Lei Zhao, Syed Ali Raza Jafri (ECE), Ethan Robert Schuchman (ECE), Hiroshi Yamauchi, Jessica Young (ECE), Dan Ardelean, Jin-Yi Wang (ECE), Chong Liang Ooi, Yen-Shiang Shue (ECE), Mohamed Gomaa (ECE), Christian Grothoff, Michael David Powell (ECE, 2004), Chen-Yong Cher (ECE, 2004), Sudipto Ghosh (2000), Ladislau-Lehel Boloni (2000), Shunge Li (2000), Steve Cutchin (1999), Ivan Krsul (1998), Gustavo Rodriguez-Rivera (1998), Muhammad Farrukh Khan (1996), Gerald Baumgartner (1996), Andrew Muckelbauer (1996), Xiangning Sean Liu (1995)

Teaching

Courses taught at Purdue

Fall 2014	CS180: Problem-Solving and Object-Oriented Programming
Spring 2014	CS456: Programming Languages
Fall 2013	CS180: Problem-Solving and Object-Oriented Programming
Fall 2012	CS352 Compilers: Principles and Practice
Spring 2012	CS661 Memory Management
Spring 2012	CS590 (CS502 equivalent): Compilers: Principles and Practice
Fall 2011	CS352 Compilers: Principles and Practice
Spring 2011	CS502 Compiling and Programming Systems
Fall 2010	CS352 Compilers: Principles and Practice
Spring 2010	CS352 Compilers: Principles and Practice
Fall 2009	CS502 Compiling and Programming Systems
Fall 2009	CS590 Principles of Programming Languages
Spring 2008	CS352 Compilers: Principles and Practice
Spring 2008	CS190C Introduction to Computational Thinking
Fall 2007	CS502 Compiling and Programming Systems
Spring 2007	CS497 Honors Project Coordinator
Spring 2007	CS197 Honors Seminar
Fall 2006	CS352 Compilers: Principles and Practice
Fall 2006	CS497 Honors Project Coordinator
Fall 2006	CS397 Honors Seminar
Spring 2006	CS497 Honors Project Coordinator
Spring 2006	CS197 Honors Seminar
Fall 2005	CS352 Compilers: Principles and Practice
Fall 2005	CS497 Honors Project Coordinator
Fall 2005	CS397 Honors Seminar
Spring 2005	CS352 Compilers: Principles and Practice
Fall 2004	CS502 Compiling and Programming Systems
Spring 2004	CS352 Compilers: Principles and Practice

Fall 2003	CS690M Advanced Dynamic Memory Management
Spring 2003	CS180 An Introduction to Computer Science
Fall 2002	CS352 Compilers: Principles and Practice
Spring 2001	CS352 Compilers: Principles and Practice
Fall 2000	CS502 Compiling and Programming Systems
Fall 2000	CS590T Software Tools
Spring 2000	CS352 Compilers: Principles and Practice
Spring 2000	CS590C Systems Software Seminar
Fall 1999	CS502 Compiling and Programming Systems
Spring 1999	CS502 Compiling and Programming Systems
Spring 1999	CS690A Advanced Compiling and Programming Systems (*or* "So you want to know how to make Java run fast?")
Fall 1998	CS502 Compiling and Programming Systems
Spring 1998	CS502 Compiling and Programming Systems
Fall 1997	CS502 Compiling and Programming Systems
Spring 1997	CS502 Compiling and Programming Systems
Fall 1996	CS502 Compiling and Programming Systems
Fall 1996	CS590D Database Programming Languages and Persistent Systems
Spring 1996	CS502 Compiling and Programming Systems
Fall 1995	CS502 Compiling and Programming Systems
Spring 1995	CS502 Compiling and Programming Systems

Courses revised at Purdue

Spring 2014	CS456 Programming Languages. Revised to use interpreter-based approach alongside textbook of Norman Ramsey
Spring 2006	CS197 Honors Seminar. Revised as a reading seminar on the canonical works of Computer Science
Fall 2003	CS180 An Introduction to Computer Science. Introduced use of Eclipse IDE into teaching
Fall 1997	CS352 Compilers: Principles and Practice. Revised curriculum and introduced new project infrastructure
Fall 1997	CS502 Compiling and Programming Systems. Revised curriculum and introduced new project infrastructure
Spring 1995	CS502 Compiling and Programming Systems. Revised curriculum and introduced new project infrastructure

Courses introduced at Purdue

Spring 2008	CS190C Introduction to Computational Thinking (with Hambrusch, Kais, Haugan, Hoffmann)
Fall 2003	CS690M Advanced Dynamic Memory Management
Spring 2000	CS590C Systems Software Seminar (with Doug Comer)

Spring 1999	CS690A Advanced Compiling and Programming Systems (*or* "So you want to know how to make Java run fast")
Fall 1996	CS590D Database Programming Languages and Persistent Systems

Institutional service

College of Science, Purdue University

2014-2016	**Member**, *Faculty Council*.
2011–2015	**Member**, *Undergraduate Curriculum and Academic Policy Committee (UCAP)*.
2004–2005	**Senior Chair**, *United Way Campaign*.
2003–2004	**Junior Chair**, *United Way Campaign*.
2000–2001	**Member**, *Grade Appeal Committee*.

Department of Computer Science, Purdue University

2014–2015	**Chair**, *Undergraduate Study Committee*.
2013–2014	**Chair**, *Undergraduate Study Committee*.
2012–2013	**Chair**, *Undergraduate Study Committee*.
2012–2013	**Member**, *Executive Committee*.
2011–2012	**Chair**, *Undergraduate Study Committee*.
2011–2012	**Member**, *Executive Committee*.
2010–2011	**Member**, *Graduate Admissions Committee*.
2009–2010	**Member**, *Awards Committee*.
2007–2008	**Member**, *Awards Committee*.
2006–2007	**Chair**, *Undergraduate Curriculum Committee*.
2006–2007	**Member**, *Executive Committee*.
2006–2007	**Advisor**, *Honors Research*.
2006–2007	**Advisor**, *Undergraduate Student Board*.
2005–2006	**Chair**, *Undergraduate Curriculum Committee*.
2005–2006	**Advisor**, *Honors Research*.
2005–2006	**Advisor**, *Undergraduate Student Board*.
2004–2005	**Member**, *Computing Facilities Committee*.
2003–2004	**Member**, *Faculty Search Committee*.
2002–2003	**Member**, *Faculty Search Committee*.
2001–present	**Member**, *Promotions Committee*, Associate and Full Professors, voting on promotions to Associate Professor with tenure.
2000–2001	**Member**, *Undergraduate Curriculum sub-Committee*.
1999–2000	**Member**, *Undergraduate Curriculum sub-Committee*.
1998–1999	**Secretary**, *Faculty Meetings*.
1997–1998	**Member**, *Undergraduate Committee*.
1997–1998	**Member**, *Executive Committee*.
1996–1997	**Member**, *Faculty Search Committee*.