

Roy, Amitabha; Hand, Steven; Harris, Tim

Weak atomicity for the x86 memory consistency model. (English) Zbl 1248.68126
J. Parallel Distrib. Comput. 72, No. 10, 1306-1317 (2012).

Summary: We consider the interaction of weakly atomic software transactional memory (STM) providing single global lock atomicity with the x86 memory consistency model. We show that a practical design for such an STM requires that some program behaviour be disallowed, due to the strictness of the x86 memory consistency model in comparison to the language level memory models hitherto considered in weakly atomic STM designs. We present the design and construction of such an STM that disallows races between a transactional read and a non-transactional write. We also report on a practical application of this STM to elide legacy locks in x86 binaries. This allows software transactional memory to be applied without requiring software to be a priori written with awareness of transactional memory and without any restriction on source language or compiler. As an example, we show how a mainstream multiplayer game can use transactional memory with zero changes and 11% overhead over language level transactional memory, which requires over 700 annotations and severely restricts software development.

MSC:

[68M99](#) Computer system organization
[68N99](#) Theory of software

Keywords:

[software transactional memory](#); [weak atomicity](#); [x86 memory consistency model](#)

Software:

[JudoSTM](#); [QuakeTM](#)

Full Text: [DOI](#)

References:

- [1] Sarita V. Adve, Mark D. Hill, Weak ordering—a new definition, in: 25 Years of the International Symposia on Computer Architecture, Selected Papers, 1998, pp. 363–375.
- [2] Intel architectures instruction set extensions programming reference, February 2012.
- [3] Cascaval, Calin; Blundell, Colin; Michael, Maged; Cain, Harold W.; Wu, Peng; Chiras, Stefanie; Chatterjee, Siddhartha: Software transactional memory: why is it only a research toy?, Queue 6, 46–58 (2008)
- [4] Dave Christie, JaeWoong Chung, Stephan Diestelhorst, Michael Hohmuth, Martin Pohlack, Christof Fetzter, Martin Nowack, Torvald Riegel, Pascal Felber, Patrick Marlier, Etienne Rivière, Evaluation of AMD’s advanced synchronization facility within a complete transactional memory stack, in: Proceedings of the European Conference on Computer Systems, 2010, pp. 27–40.
- [5] Dave Dice, Ori Shalev, Nir Shavit, Transactional locking II, in: Proceedings of the International Symposium on Distributed Computing, 2006, pp. 194–208.
- [6] Keir Fraser, Practical lock freedom, Ph.D. Thesis, Cambridge University Computer Laboratory, 2003. Also available as: Technical Report UCAM-CL-TR-579.
- [7] Vladimir Gajinov, Ferad Zyulkyarov, Osman S. Unsal, Adrian Cristal, Eduard Ayguade, Tim Harris, Mateo Valero, QuakeTM: parallelizing a complex sequential application using transactional memory, in: Proceedings of the International Conference on Supercomputing, 2009, pp. 126–135.
- [8] Rachid Guerraoui, Michal Kapalka, On the correctness of transactional memory, in: Proceedings of the Symposium on Principles and Practice of Parallel Programming, 2008, pp. 175–184. · [Zbl 1315.68065](#)
- [9] Tim Harris, Keir Fraser, Language support for lightweight transactions, in: Proceedings of the Conference on Object-Oriented Programming, Systems, Languages, and Applications, 2003, pp. 388–402.
- [10] Harris, Tim; Larus, Jim; Rajwar, Ravi: Transactional memory, (2010)
- [11] Jeremy Manson, William Pugh, Sarita V. Adve, The Java memory model, in: POPL’05, 32nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, 2005, pp. 378–391. · [Zbl 1369.68079](#)
- [12] Vijay Menon, Steven Balensiefer, Tatiana Shpeisman, Ali-Reza Adl-Tabatabai, Richard L. Hudson, Bratin Saha, Adam Welc,

Towards a lock-based semantics for Java STM, Technical Report UW-CSE-07-11-01, University of Washington, November 2007.

- [13] Vijay Menon, Steven Balensiefer, Tatiana Shpeisman, Ali-Reza Adl-Tabatabai, Richard L. Hudson, Bratin Saha, Adam Welc, Practical weak-atomicity semantics for Java STM, in: Proceedings of the Symposium on Parallelism in Algorithms and Architectures, 2008, pp. 314–325.
- [14] Vijay Menon, Steven Balensiefer, Tatiana Shpeisman, Ali-Reza Adl-Tabatabai, Richard Hudson, Bratin Saha, Adam Welc, Single global lock semantics in a weakly atomic STM, in: Proceedings of the ACM SIGPLAN Workshop on Languages, Compilers, and Hardware Support for Transactional Computing, 2008.
- [15] Takuya Nakaike, Maged M. Michael, Lock elision for read-only critical sections in Java, in: Proceedings of the Conference on Programming Language Design and Implementation, 2010, pp. 269–278.
- [16] Marek Olszewski, Jeremy Cutler, J. Gregory Steffan, JudoSTM: a dynamic binary-rewriting approach to software transactional memory, in: Proceedings of the International Conference on Parallel Architecture and Compilation Techniques, 2007, pp. 365–375.
- [17] Scott Owens, Susmit Sarkar, Peter Sewell, A better x86 memory model: x86-TSO, in: Proceedings of the Conference on Theorem Proving in Higher Order Logics, 2009.
- [18] Paruj Ratanaworabhan, Martin Burtscher, Darko Kirovski, Benjamin Zorn, Rahul Nagpal, Karthik Pattabiraman, Detecting and tolerating asymmetric races, in: Proceedings of the Symposium on Principles and Practice of Parallel Programming, 2009, pp. 173–184.
- [19] Amitabha Roy, Software lock elision for x86 machine code, Technical Report UCAM-CL-TR-801, University of Cambridge, Computer Laboratory, 2011.
- [20] Amitabha Roy, Steven Hand, Tim Harris, A runtime system for software lock elision, in: Proceedings of the European Conference on Computer Systems, 2009, pp. 261–274.
- [21] Takayuki Usui, Yannis Smaragdakis, Reimer Behrends, Adaptive locks: combining transactions and locks for efficient concurrency, in: Proceedings of the International Conference on Parallel Architectures and Compilation Techniques, 2009. · [Zbl 1233.68172](#)
- [22] Cheng Wang, Wei-Yu Chen, Youfeng Wu, Bratin Saha, Ali-Reza Adl-Tabatabai, Code generation and optimization for transactional memory constructs in an unmanaged language, in: Proceedings of the International Symposium on Code Generation and Optimization, 2007, pp. 34–48.
- [23] Lukasz Ziarek, Adam Welc, Ali-Reza Adl-Tabatabai, Vijay Menon, Tatiana Shpeisman, Suresh Jagannathan, A uniform transactional execution environment for Java, in: Proceedings of the European Conference on Object-Oriented Programming, 2008, pp. 129–154.

This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.