

The GC Interface in the EVM

Derek White and Alex Garthwaite

Introduction by Steve Heller

This version of "The GC Interface in the EVM" was shortened by Alex Garthwaite, and includes an updated "Addenda, Open Issues, and Future Directions" section. Visit www.sun.com/research/jtech/ for the full version.

The Java™ Technology Research Group (originally known as Java Topics) was formed late in 1995 to develop expertise in Java garbage collection (GC). The founding members of the group, Guy Steele, Derek White and Steve Heller, were joined by Ole Agesen during the summer of 1996, and we set the goal to explore soft real-time GC, probabilistically meeting real time constraints. We wanted to "solve the Java GC problem," but we soon realized that Java didn't *have* a GC problem. Java implementations at the time were so slow that GC was not the weakest link. We set out to build a high-performance Java™ Virtual Machine (JVM) that would have a GC problem so that we could solve it. And we carefully designed this JVM to facilitate GC experimentation.

Soon after Christine Flood and Dave Detlefs joined the team in the fall of 1996, Ole Agesen led an effort to overhaul the Sun JVM™ now known as the "Classic VM" to create a JVM known as "ExactVM" (also known as "EVM," and eventually known as "The Sun Microsystems Laboratories Virtual Machine for Research," or "ResearchVM"). At the center of the ExactVM was a GC interface, designed to facilitate GC experimentation. Alex Garthwaite, who joined the team as a summer intern in 1997 and 1998 and a full time member in the fall of 1998, teamed up with Derek White to produce the GC interface document (this TR).

We also joined forces with a product team whose goal was to build a high-performance JVM for the Solaris™ Operating Environment. The GC interface abstraction did not cost the VM performance, and ExactVM became Sun's Java™ 2 SDK Production Release for Solaris OE, versions 1.2.2_xx, managed by David Nelson-Gal, Borek Vokach-Brodsky, Dave Seberger (and many others), led technically by Ole Agesen and Ross Knippel (Mario Wolczko was the initial product team technical lead), and contributed to by scores of others. Three years later, the ExactVM is still packaged with Solaris and shipped as a product.

The ExactVM team was a close and highly successful collaboration between research and development; together we investigated various aspects of Java performance and scalability, notably synchronization, just-in-time compilation, and garbage collection. This project was a success disaster as the research team was drawn tightly into the product team. David Nelson-Gal, the director of the product side of the team, extracted a written commitment (dated 8/26/97) from the research side of the team to remain committed to the project for at least one year. We did that and more.

The GC interface supported various generational garbage collection configurations, the default system being a semi-spaces young generation and a mark-compact mature generation. We also developed many experimental GC configurations. As is often the case, we leveraged the momentum and energy of summer interns: during the summer of 1998 Alex Garthwaite and Tony Printezis, respectively, designed and implemented incremental (a bit at a time) and concurrent (simultaneous with the Java program) collectors. The next summer we also explored parallel (multi-threaded, led by Christine Flood, with help from summer intern Catherine Zhang) GC. None of these technologies were made widely available to customers in 1.2.2_xx.

Within Sun, at least one other product group and a host of other research teams leveraged the GC interface and the ExactVM to experiment or to build other JVMs. On the product side, Borek Vokach–Brodsky and his team built the J2ME™ Connected Device Configuration which contains the CVM, a small footprint JVM based on ExactVM and the GC interface. The following efforts were on the research side. Mario Wolczko derived the "Tracing JVM" for extracting and studying Java traces, and it was licensed to nine universities and several companies. Mick Jordan, Malcolm Atkinson (Glasgow University), and their teams derived a JVM that implemented orthogonal persistence for the Java platform by leveraging the GC interface; this JVM was also licensed to a number of universities. In a related project Laurent Daynes used the ExactVM extensively to experiment with transaction–based isolation between applications. Grzegorz Czajkowski and Laurent used the ExactVM to evaluate the performance of a multi–tasking JVM prototype based on bytecode editing and to test the native method isolation mechanism.

ExactVM source was licensed to a number of academic institutions to facilitate research, especially in garbage collection: UC Santa Barbara, Glasgow University, Purdue, Tokyo Institute of Technology, Pennsylvania State University, Rice University, University of Kent, University of Waterloo, University of Texas at Austin, Technion, Princeton, University of Cambridge, San Francisco State University, and the University of Toronto. The success of this program was due in large part to the GC interface document, which was the only external documentation delivered with the ExactVM source. The success can be measured by the many papers and theses enabled by the program, some of which are listed below and some of which are yet to emerge.

A second version of the GC Interface was designed by Dave Detlefs for use in the Java HotSpot™ JVM and is part of Sun's 1.4 release. In subsequent releases, we expect that many of the incremental, concurrent, and parallel technologies we developed will be deployed in products. This represents another close and successful collaboration between research and development, led technically by Dave Detlefs on the research side, and Peter Kessler on the product side.

The current research of the team continues to build on the incremental, concurrent, and parallel techniques developed in conjunction with the GC interface and ExactVM. We walk the hairy edge between research and development.

While I've attempted to retell the story of our GC interface and the ExactVM (they are inseparable), it was such a broad team effort that I undoubtedly have not given adequate credit in some cases. We worked with many researchers and product people here at Sun. We worked with interns, visiting professors, and academic collaborators. And, although I will make him uncomfortable by concluding thus, the undisputed champion and leader of our effort was Ole Agesen, to whom we owe a great debt. [note to self: send Ole a certificate of appreciation (use color printer) and a couple of pounds of "Peet's"]

RELATED GROUP PUBLICATIONS (see <http://www.sun.com/research/jtech/>):

Ole Agesen. 1998 "GC Points in a Threaded Environment." Sun Labs TR 98–70. Mountain View, CA

Ole Agesen. 1999 "Space and Time–Efficient Hashing of Garbage–Collected Objects." Theory and Practice of Object Systems, 5(2) (June): 119–124.

Ole Agesen and David Detlefs. 1997. "Finding References in Java(TM) Stacks." OOPSLA Workshop on Garbage Collection and Memory Management. Atlanta, GA

Ole Agesen, David Detlefs, Alex Garthwaite, Ross Knippel, Y.S. Ramakrishna, Derek White. 1999. "An Efficient Meta-lock for Implementing Ubiquitous Synchronization." Sun Labs TR 99-76. Mountain View, CA. Presented at OOPSLA '99.

Ole Agesen, David Detlefs, and J. Eliot B. Moss. 1998. "Garbage Collection and Local Variable Type-Precision and Liveness in Java(TM) Virtual Machines." ACM SIGPLAN Conference on Programming Language Design and Implementation (June):269-279. Montreal, Canada

Ole Agesen and Alex Garthwaite. 2000. "Efficient Object Sampling Via Weak References." ISMM2000. Minneapolis, MN

Dave Detlefs and Tony Printezis. 2000. "A Generational Mostly-Concurrent Garbage Collector" Sun Labs TR-2000-88. Mountain View, CA. Short version appeared in ISMM2000.

Christine Flood, Dave Detlefs, Nir Shavit, Catherine Zhang. 2001. "Parallel Garbage Collection for Shared Memory Multiprocessors." Java Virtual Machine Research and Technology Symposium. Monterey, CA

Timothy L. Harris. 2000. "Dynamic Adaptive Pre-Tenuring." ISMM2000. Minneapolis, MN

Derek White and Alex Garthwaite. 1998. "The GC Interface in the EVM." Sun Labs TR 98-67. Mountain View, CA

RELATED OUTSIDE PUBLICATIONS (GENERALLY ENABLED BY THE SOURCE LICENSING PROGRAM)

M. P. Atkinson, M. Dmitriev, C. G. Hamilton, T. Printezis. 2000. "Scalable and Recoverable Implementation of Object Evolution for the PJama_1 Platform," Proceedings of The Ninth International Workshop on Persistent Object Systems (POS9). Lillehammer, Norway.

M. P. Atkinson and M. J. Jordan 2000. "A Review of the Rationale and Architectures of PJama: A Durable, Flexible, Evolvable and Scalable Orthogonally Persistent Programming Platform," Sun Labs TR-2000-90.

Kumar Brahmamath. May 1998. "Optimizing Orthogonal Persistence for Java," Master's Thesis, Purdue University.

Kumar Brahmamath, Nathaniel Nystrom, Antony L. Hosking, Quintin Cutts. 1998, 1999. "Swizzle barrier optimizations for orthogonal persistence in Java," In Proceedings of the Third International Workshop on Persistence and Java (Tiburon, California, September 1998). Published as Advances in Persistent Object Systems, Morrison, Jordan and Atkinson (Eds.), pp 268-278. Morgan Kaufmann, 1999

Quintin Cutts and Antony L. Hosking. 1997. "Analysing, profiling and optimising orthogonal persistence for Java," In Proceedings of the Second International Workshop on Persistence and Java (Half Moon Bay, California, August 1997). Sun Microsystems Laboratories Technical Report 97-63, pp 107-115.

Quintin Cutts, Steve Lennon and Antony Hosking, 1998, 1999. "Reconciling buffer management with persistence optimisations," In Eighth International Workshop on Persistent Object Systems" (Tiburon, California, August 1998). Published as Advances in Persistent Object Systems, Morrison, Jordan and Atkinson (Eds.), pp 51-63. Morgan Kaufmann, 1999.

Sylvia Dieckmann and Urs Hoelzle. 1999. "A Study of the Allocation Behavior of the SPECjvm98 Java Benchmarks." In Proceedings of the 13th European Conference on Object-Oriented Programming (ECOOP'99), Lisbon, Springer Verlag.

Sylvia Dieckmann and Urs Hoelzle. 1999. A Case for Using Active Memory to Support Garbage Collection. Workshop on Hardware Support for Objects and Microarchitectures in Java. Held in Conjunction with ICCD'99, Austin, TX

Antony Hosking, Nathaniel Nystrom, Quintin Cutts and Kumar Brahmamath. 1998, 1999. "Optimizing the read and write barrier for orthogonal persistence," In Eighth International Workshop on Persistent Object Systems (Tiburon, California, August 1998). Published as Advances in Persistent Object Systems, Morrison, Jordan and Atkinson (Eds.), pp 149–159. Morgan Kaufmann, 1999.

Antony L. Hosking, Nathaniel Nystrom, David Whitlock, Quintin Cutts and Amer Diwan. 2001. "Partial redundancy elimination for access path expressions." Software – Practice and Experience 31(6):577–600, 2001.

E. Hunt, M. P. Atkinson, and R. W. Irving. 2001. "A Database Index To Large Biological Sequences," Proceedings of the 27th International Conference on Very Large Databases (VLDB 2001). Morgan Kaufmann, Roma, Italy.

Nathaniel Nystrom. 1998. "Bytecode Level Analysis and Optimization of Java Classes," Master's Thesis, Purdue University.

T. Printezis. 2000 . "Management of Long-Running High-Performance Persistent Object Stores," PhD Thesis. Department of Computing Science, University of Glasgow, Scotland.

Tony Printezis. 2001. "Hot-Swapping Between a Mark&Sweep and a Mark&Compact Garbage Collector in a Generational Environment." Java Virtual Machine Research and Technology Symposium. Monterey, CA. Also: TR-2001-78, Department of Computing Science, University of Glasgow, Scotland.

T. Printezis and M. P. Atkinson. 2001. "An Efficient Object Promotion Algorithm for Persistent Object Systems," SPE 31(10) 941--981.

Darko Stefanovic. 1999. "Properties of Age-Based Automatic Memory Reclamation Algorithms" Doctoral Dissertation, University of Massachusetts Amherst.

Darko Stefanovic, J. Eliot B. Moss, and Kathryn S. McKinley. 1999. "Age-Based Garbage Collection," Proceedings of the ACM Conference on Object-Oriented Programming Systems, Languages, and Applications, Denver, Colorado.

David Whitlock and Antony L. Hosking. 2000. "A Framework for Persistence-Enabled Optimization of Java Object Stores," In Ninth International Workshop on Persistent Object Systems." Lillehammer, Norway.

David Whitlock. 2000. "Persistence-enabled optimization of Java programs," Master's Thesis, Purdue University.