

























- [6] B. Dufour, C. Goard, L. Hendren, O. de Moor, G. Sittampalam, and C. Verbrugge. Measuring the dynamic behaviour of AspectJ programs. In *Proc. OOPSLA 2004*, pages 150–169, New York, NY, USA, 2004. ACM.
- [7] J. Ehlers, A. van Hoorn, J. Waller, and W. Hasselbring. Self-adaptive software system monitoring for performance anomaly localization. In *Proc. ICAC 2011*, pages 197–200, Karlsruhe, Germany, June 2011. ACM.
- [8] H. Eichelberger and K. Schmid. Flexible resource monitoring of Java programs. *Journal of Systems and Software*, 93:163–186, July 2014.
- [9] K. Govindraj, S. Narayanan, B. Thomas, P. Nair, and S. Peeru. On using AOP for application performance management. In *Proc. AOSD 2006*, pages 18–30, 2006.
- [10] M. Grimmer, M. Rigger, L. Stadler, R. Schatz, and H. Mössenböck. An efficient native function interface for Java. In *Proc. PPPJ 2013*, pages 35–44, New York, NY, USA, 2013. ACM.
- [11] M. Haupt and M. Mezini. Micro-measurements for dynamic aspect-oriented systems. In *Object-Oriented and Internet-Based Technologies*, number 3263 in LNCS, pages 81–96. Springer Berlin Heidelberg, 2004.
- [12] V. Horký, P. Libič, L. Marek, A. Steinhauser, and P. Tůma. Utilizing performance unit tests to increase performance awareness. In *Proc. ICPE 2015*, pages 289–300, New York, NY, USA, 2015. ACM.
- [13] T. Kalibera, L. Bulej, and P. Tuma. Benchmark precision and random initial state. In *Proc. SPECTS 2005*, pages 853–862. SCS, 2005.
- [14] T. Kotzmann, C. Wimmer, H. Mössenböck, T. Rodriguez, K. Russell, and D. Cox. Design of the java hotspot client compiler for java 6. *ACM Transactions on Architecture and Code Optimization*, 5(1):7:1–7:32, May 2008.
- [15] D. Lea. The JSR-133 cookbook for compiler writers, 2011. <http://gee.cs.oswego.edu/dl/jmm/cookbook.html>.
- [16] P. Libič, L. Bulej, V. Horký, and P. Tůma. On the limits of modeling generational garbage collector performance. In *Proc. ICPE 2014*, pages 15–26, New York, NY, USA, 2014. ACM.
- [17] P. Libič, L. Bulej, V. Horký, and P. Tůma. Estimating the impact of code additions on garbage collection overhead. In *Proc. EPEW 2015*, number 9272 in LNCS, pages 130–145. Springer International Publishing, Aug. 2015.
- [18] A. D. Malony. *Performance Observability*. PhD thesis, University of Illinois at Urbana-Champaign, Champaign, IL, USA, 1990. AAI9114332.
- [19] A. D. Malony and S. S. Shende. Overhead compensation in performance profiling. In *Proc. Euro-Par 2004*, number 3149 in LNCS, pages 119–132. Springer Berlin Heidelberg, Aug. 2004.
- [20] A. D. Malony and S. S. Shende. Models for on-the-fly compensation of measurement overhead in parallel performance profiling. In *Proc. Euro-Par 2005*, number 3648 in LNCS, pages 72–82. Springer Berlin Heidelberg, Aug. 2005.
- [21] L. Marek, A. Villazón, Y. Zheng, D. Ansaloni, W. Binder, and Z. Qi. DiSL: A domain-specific language for bytecode instrumentation. In *Proc. AOSD 2012*, pages 239–250, New York, NY, USA, 2012. ACM.
- [22] T. Martinec, L. Marek, A. Steinhauser, P. Tůma, Q. Noorshams, A. Rentschler, and R. Reussner. Constructing performance model of JMS middleware platform. In *Proc. ICPE 2014*, pages 123–134, New York, NY, USA, 2014. ACM.
- [23] A. Mos and J. Murphy. COMPAS: Adaptive performance monitoring of component-based systems. In *Proc. ICSE 2004 RAMSS*, 2004.
- [24] Oracle. Java microbenchmark harness, 2013-2015. <http://openjdk.java.net/projects/code-tools/jmh>.
- [25] P. Panchamukhi. Kernel debugging with kprobes, 2004. <http://www.ibm.com/developerworks/library/l-kprobes/index.html>.
- [26] N. Park, B. Hong, and V. Prasanna. Tiling, block data layout, and memory hierarchy performance. *IEEE Transactions on Parallel and Distributed Systems*, 14(7):640–654, July 2003.
- [27] T. Parsons, A. Mos, and J. Murphy. Non-intrusive end-to-end runtime path tracing for J2EE systems. *Software, IEEE Proceedings*, 153(4):149–161, Aug. 2006.
- [28] D. J. Pearce, M. Webster, R. Berry, and P. H. J. Kelly. Profiling with AspectJ. *Software: Practice and Experience*, 37(7):747–777, June 2007.
- [29] A. Sarimbekov, A. Sewe, W. Binder, P. Moret, and M. Mezini. JP2: Call-site aware calling context profiling for the Java Virtual Machine. *Science of Computer Programming*, 79:146–157, Jan. 2014.
- [30] S. S. Shende and A. D. Malony. The Tau parallel performance system. *Int. J. High Perform. Comput. Appl.*, 20(2):287–311, May 2006.
- [31] A. Shipilëv. Java Microbenchmark Harness (The Lesser of Two Evils). Presentation at Devovx, 2013. <http://shipilev.net/talks/devovx-Nov2013-benchmarking.pdf>.
- [32] SPASS-meter monitoring framework, 2015. <http://www.sse.uni-hildesheim.de/spass-meter>.
- [33] SPEC Java server business benchmark, 2015. <http://www.spec.org/jbb2015>.
- [34] A. Tamches and B. P. Miller. Fine-grained dynamic instrumentation of commodity operating system kernels. In *Proc. OSDI 1999*, pages 117–130, Berkeley, CA, USA, 1999. USENIX Association.
- [35] A. van Hoorn, M. Rohr, W. Hasselbring, J. Waller, J. Ehlers, S. Frey, and D. Kieselhorst. Continuous monitoring of software services: Design and application of the Kieker framework. Report, Department of Computer Science, Kiel University, Germany, Nov. 2009.
- [36] A. van Hoorn, J. Waller, and W. Hasselbring. Kieker: A framework for application performance monitoring and dynamic software analysis. In *Proc. ICPE 2012*, pages 247–248, Boston, Massachusetts, USA, Apr. 2012. ACM.
- [37] J. Waller, F. Fittkau, and W. Hasselbring. Application performance monitoring: Trade-off between overhead reduction and maintainability. In *Proceedings of the Symposium on Software Performance 2014*, pages 1–24, Stuttgart, Germany, Nov. 2014. University of Stuttgart.