

REFERENCES

- [1] “Java Mission Control.” [Online]. Available: <https://www.oracle.com/technetwork/java/javaseproducts/mission-control/index.html>. [Accessed: 16-May-2019].
- [2] “Apache JMeter™.” [Online]. Available: <https://jmeter.apache.org/>. [Accessed: 16-May-2019].
- [3] M. Shaw and P. Clements, “A field guide to boxology: Preliminary classification of architectural styles for software systems,” in *Proceedings Twenty-First Annual International Computer Software and Applications Conference (COMPSAC’97)*, 1997, pp. 6–13.
- [4] M. Welsh, D. Culler, and E. Brewer, “SEDA: an architecture for well-conditioned, scalable internet services,” in *ACM SIGOPS Operating Systems Review*, 2001, vol. 35, no. 5, pp. 230–243.
- [5] P. Haller and M. Odersky, “Scala actors: Unifying thread-based and event-based programming,” *Theor. Comput. Sci.*, vol. 410, no. 2–3, pp. 202–220, 2009.
- [6] “JEP 107: Bulk Data Operations for Collections.” [Online]. Available: <http://openjdk.java.net/jeps/107>. [Accessed: 18-May-2019].
- [7] “Scala 2.8 Collections API.” [Online]. Available: <https://www.scala-lang.org/docu/files/collections-api/collections.html>. [Accessed: 18-May-2019].
- [8] “ReactiveX, ‘ReactiveX/RxJava,’ GitHub.” [Online]. Available: <https://github.com/ReactiveX/RxJava>. [Accessed: 18-May-2019].
- [9] H. Gomaa, “Catalog of Software Architectural Patterns,” in *Software Modeling and Design*, pp. 495–520.
- [10] “Web on Reactive Stack.” [Online]. Available: <https://docs.spring.io/spring/docs/current/spring-framework-reference/web-reactive.html>. [Accessed: 19-May-2019].
- [11] “The Java Community Process(SM) Program - JSRs: Java Specification Requests - detail JSR# 163.” [Online]. Available: <https://www.jcp.org/en/jsr/detail?id=163>. [Accessed: 19-May-2019].
- [12] M. Gagliardi and Y. Sun, “Conditional dynamic instrumentation of software in a specified transaction context.” Google Patents, 2013.
- [13] M. Dmitriev, “Design of JFluid: A profiling technology and tool based on dynamic bytecode instrumentation,” 2003.

- [14] P. Abercrombie and M. Karaorman, “jContractor: Bytecode instrumentation techniques for implementing design by contract in Java,” *Electron. Notes Theor. Comput. Sci.*, vol. 70, no. 4, pp. 55–79, 2002.
- [15] J. Whaley, “A portable sampling-based profiler for Java virtual machines,” in *Proceedings of the ACM 2000 conference on Java Grande*, 2000, pp. 78–87.
- [16] M. Rohr *et al.*, “Kieker: Continuous monitoring and on demand visualization of Java software behavior,” 2008.
- [17] ““JSR 51: New I/O APIs for the Java™ Platform,” The Java Community Process(SM) Program - JSRs: Java Specification Requests - detail JSR# 51.” [Online]. Available: <https://www.jcp.org/en/jsr/detail?id=51>. [Accessed: 16-May-2019].
- [18] ““JSR 203: More New I/O APIs for the Java™ Platform (“NIO.2”),’ The Java Community Process(SM) Program - JSRs: Java Specification Requests - detail JSR# 203.” [Online]. Available: <https://www.jcp.org/en/jsr/detail?id=203>. [Accessed: 16-May-2019].
- [19] M. Welsh and D. E. Culler, “Adaptive Overload Control for Busy Internet Servers.,” in *USENIX Symposium on Internet Technologies and Systems*, 2003, p. 4.
- [20] S. Ertel, C. Fetzer, and P. Felber, “Ohua: Implicit Dataflow Programming for Concurrent Systems,” in *Proceedings of the Principles and Practices of Programming on The Java Platform*, 2015, pp. 51–64.
- [21] “Clojure.” [Online]. Available: <https://clojure.org/>. [Accessed: 16-May-2019].
- [22] E. Kiciman, B. Livshits, M. Musuvathi, and K. C. Webb, “Fluxo: a system for internet service programming by non-expert developers,” in *Proceedings of the 1st ACM symposium on Cloud computing*, 2010, pp. 107–118.
- [23] G. A. Agha, “Formal methods for Actor systems: A progress report,” in *Proceedings of the IFIP TC6/WG6. 1 5th International Conference on Formal Description Techniques for Distributed Systems and Communications Protocols-FORTE’92*, 1993, pp. 217–228.
- [24] T. Van Cutsem, S. Mostinckx, and W. De Meuter, “Linguistic symbiosis between event loop actors and threads,” *Comput. Lang. Syst. Struct.*, vol. 35, no. 1, pp. 80–98, 2009.
- [25] ““Akka: build concurrent, distributed, and resilient message-driven applications for Java and Scala | Akka.”” [Online]. Available: <https://akka.io/>. [Accessed: 16-May-2019].
- [26] D. Pariag, T. Brecht, A. Harji, P. Buhr, A. Shukla, and D. R. Cheriton, “Comparing the performance of web server architectures,” in *ACM SIGOPS Operating Systems Review*, 2007, vol. 41, no. 3, pp. 231–243.
- [27] M. Hirzel, R. Soulé, S. Schneider, B. Gedik, and R. Grimm, “A catalog of stream

- processing optimizations,” *ACM Comput. Surv.*, vol. 46, no. 4, p. 46, 2014.
- [28] B. Gedik, H. G. Özsema, and Ö. Öztürk, “Pipelined fission for stream programs with dynamic selectivity and partitioned state,” *J. Parallel Distrib. Comput.*, vol. 96, pp. 106–120, 2016.
- [29] M. Thompson, D. Farley, M. Barker, P. Gee, and A. Stewart, “Disruptor: High performance alternative to bounded queues for exchanging data between concurrent threads,” *Tech. Pap. LMAX, May*, p. 206, 2011.
- [30] “‘spring.io,’ Spring.” [Online]. Available: <https://spring.io/>. [Accessed: 19-May-2019].
- [31] “‘Java Management Extensions (JMX) Technology,’ Java Management Extensions (JMX).” [Online]. Available: <https://www.oracle.com/technetwork/java/javase/tech/javamanagement-140525.html>. [Accessed: 21-May-2019].
- [32] “‘React – A JavaScript library for building user interfaces,’ – A JavaScript library for building user interfaces.” [Online]. Available: <https://reactjs.org/>. [Accessed: 21-May-2019].
- [33] “MySQL.” [Online]. Available: <https://www.mysql.com/>. [Accessed: 21-May-2019].
- [34] “‘ETSI TS 124 390 V11.0.0 : Unstructured Supplementary Service Data (USSD) specification.,’ ETSI TS 124 390 V11.0.0 : Unstructured Supplementary Service Data (USSD) specification.” [Online]. Available: https://www.etsi.org/deliver/etsi_ts/124300_124399/124390/11.00.00_60/ts_124390v110000p.pdf. [Accessed: 21-May-2019].
- [35] “‘WireMock,’ WireMock.” [Online]. Available: <http://wiremock.org/>. [Accessed: 21-May-2019].