

List of Citations

Lubomír Bulej

November 2018

The list is based on information from common publication databases. Following the Charles University rector directives 9/2014 and 17/2014, the list contains citations from monographs, scientific periodicals and reviewed proceedings, excluding self citations where the applicant is among the authors of the citing publication. Also excluded are all citing works originating wholly from the applicant's department, foreign language publications (because this makes determining the nature of the citation difficult) and theses (because it is not clear whether theses qualify as monographs). The value of H-index based on the citing works listed here is 8.

[1] L. Bulej, T. Kalibera, and P. Tůma: **“Repeated Results Analysis for Middleware Regression Benchmarking”**. In *Performance Evaluation* 60.1–4 (2005), pp. 345–358. DOI: 10.1016/j.peva.2004.10.013. WOS 2005 IF 0.756, SCOPUS 2005 SNIP 1.568

1. K. Angerbauer, D. Okanović, A. van Hoorn, and C. Heger: **“The Back End Is Only One Part of the Picture: Mobile-Aware Application Performance Monitoring and Problem Diagnosis”**. In *Proc. VALUETOOLS*. ACM, 2017, pp. 82–89. DOI: 10.1145/3150928.3150939. SCOPUS
2. A. Brunnert and H. Krcmar: **“Continuous Performance Evaluation and Capacity Planning Using Resource Profiles for Enterprise Applications”**. In *Journal of Systems and Software* 123 (2017), pp. 239–262. DOI: 10.1016/j.jss.2015.08.030. WOS, SCOPUS
3. C. Heger, A. V. Hoorn, D. Okanović, S. Siegl, and A. Wert: **“Expert-Guided Automatic Diagnosis of Performance Problems in Enterprise Applications”**. In *Proc. EDCC*. 2016, pp. 185–188. DOI: 10.1109/EDCC.2016.16. SCOPUS
4. C. Vögele, A. van Hoorn, E. Schulz, W. Hasselbring, and H. Krcmar: **“WESSBAS: Extraction of Probabilistic Workload Specifications for Load Testing and Performance Prediction—a Model-Driven Approach for Session-Based Application Systems”**. In *Software & Systems Modeling* (2016), pp. 1–35. DOI: 10.1007/s10270-016-0566-5. SCOPUS
5. Q. Luo, D. Poshyvanyk, and M. Grechanik: **“Mining Performance Regression Inducing Code Changes in Evolving Software”**. In *Proc. MSR*. ACM, 2016, pp. 25–36. DOI: 10.1145/2901739.2901765. SCOPUS
6. K. C. Foo, Z. M. Jiang, B. Adams, A. E. Hassan, Y. Zou, and P. Flora: **“An Industrial Case Study on the Automated Detection of Performance Regressions in Heterogeneous Environments”**. In *Proc. ICSE*. 2015, pp. 159–168. DOI: 10.1109/ICSE.2015.144. SCOPUS
7. Z. M. Jiang and A. E. Hassan: **“A Survey on Load Testing of Large-Scale Software Systems”**. In *IEEE Transactions on Software Engineering* PP.99 (2015), p. 32. DOI: 10.1109/TSE.2015.2445340. SCOPUS
8. A. Wert, M. Oehler, C. Heger, and R. Farahbod: **“Automatic Detection of Performance Anti-Patterns in Inter-Component Communications”**. In *Proc. QOSA*. ACM, 2014, pp. 3–12. DOI: 10.1145/2602576.2602579. SCOPUS
9. C. Heger, J. Happe, and R. Farahbod: **“Automated Root Cause Isolation of Performance Regressions During Software Development”**. In *Proc. ICPE*. ACM, 2013, pp. 27–38. DOI: 10.1145/2479871.2479879. SCOPUS
10. A. Wert: **“Performance Problem Diagnostics by Systematic Experimentation”**. In *Proc. WCOP*. ACM, 2013, pp. 1–6. DOI: 10.1145/2465498.2465499. SCOPUS
11. A. Wert, J. Happe, and L. Happe: **“Supporting Swift Reaction: Automatically Uncovering Performance Problems by Systematic Experiments”**. In *Proc. ICSE*. IEEE Press, 2013, pp. 552–561. ISBN: 978-1-4673-3076-3. WOS
12. Z. M. Jiang, A. E. Hassan, G. Hamann, and P. Flora: **“Automated Performance Analysis of Load Tests”**. In *Proc. ICSM*. 2009, pp. 125–134. DOI: 10.1109/ICSM.2009.5306331. WOS, SCOPUS
13. S. Becker, M. Trifu, and R. Reussner: **“Towards Supporting Evolution of Service-Oriented Architectures Through Quality Impact Prediction”**. In *Proc. ASE Workshops*. 2008, pp. 77–81. DOI: 10.1109/ASEW.2008.4686297. WOS, SCOPUS

14. M. Rohr, A. van Hoorn, S. Giesecke, J. Matevska, W. Hasselbring, and S. Alekseev: “**Trace-Context Sensitive Performance Profiling for Enterprise Software Applications**”. In *Proc. SIPEW*. LNCS 5119. Springer, 2008, pp. 283–302. DOI: 10.1007/978-3-540-69814-2_18. WOS, SCOPUS
15. D. Fiedler, K. Walcott, T. Richardson, G. M. Kapfhammer, A. Amer, and P. K. Chrysanthis: “**Towards the Measurement of Tuple Space Performance**”. In *SIGMETRICS Performance Evaluation Review* 33.3 (2005), pp. 51–62. DOI: 10.1145/1111572.1111574. SCOPUS

Total: 15

[2] A. Podzimek, L. Bulej, L. Y. Chen, W. Binder, and P. Tůma: “**Analyzing the Impact of CPU Pinning and Partial CPU Loads on Performance and Energy Efficiency**”. In *Proc. 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*. IEEE, 2015, pp. 1–10. DOI: 10.1109/CCGrid.2015.164. SCOPUS, CORE 2014 A, Acceptance 69 of 268 (25.7%), Best Paper Runner-Up Award

16. C. Eibel, T.-N. Do, R. Meissner, and T. Distler: “**Empya: Saving Energy in the Face of Varying Workloads**”. In *Proc. IC2E*. 2018, pp. 134–140. DOI: 10.1109/IC2E.2018.00036
17. M. Hofmann, R. Kiesel, and G. Rünger: “**Energy and Performance Analysis of Parallel Particle Solvers from the ScaFaCoS Library**”. In *Proc. ICPE*. ACM, 2018, pp. 88–95. DOI: 10.1145/3184407.3184409
18. K. N. Khan, S. Scepanovic, T. Niemi, J. K. Nurminen, S. V. Alfthan, and O.-P. Lehto: “**Analyzing the Power Consumption Behavior of a Large Scale Data Center**”. In *Computer Science - Research and Development* (2018), pp. 1–10. DOI: 10.1007/s00450-018-0394-7
19. J. Krzywda, A. Ali-Eldin, T. E. Carlson, P.-O. Östberg, and E. Elmroth: “**Power-Performance Trade-offs in Data Center Servers: DVFS, CPU Pinning, Horizontal, and Vertical Scaling**”. In *Future Generation Computer Systems* 81 (2018), pp. 114–128. DOI: 10.1016/j.future.2017.10.044
20. W. Lund and J. Lilius: “**Inactivity Benchmarking**”. In *Proc. HPCS*. 2018, pp. 1028–1033. DOI: 10.1109/HPCS.2018.00161
21. F. Pascual and K. Rządca: “**Colocating Tasks in Data Centers Using a Side-Effects Performance Model**”. In *European Journal of Operational Research* 268.2 (2018), pp. 450–462. DOI: 10.1016/j.ejor.2018.01.046
22. C. Prakash, P. Prashanth, U. Bellur, and P. Kulkarni: “**Deterministic Container Resource Management in Derivative Clouds**”. In *Proc. IC2E*. 2018, pp. 79–89. DOI: 10.1109/IC2E.2018.00030
23. F. Pascual and K. Rządca: “**Optimizing Egalitarian Performance in the Side-Effects Model of Colocation for Data Center Resource Management**”. In *Proc. Euro-Par*. Springer, 2017, pp. 206–219. DOI: 10.1007/978-3-319-64203-1_15
24. C. Prakash, Prashanth, P. Kulkarni, and U. Bellur: “**Mitigating Nesting-Agnostic Hypervisor Policies in Derivative Clouds**”. In *Proc. ICDCS*. 2017, pp. 2620–2621. DOI: 10.1109/ICDCS.2017.280
25. S. A. R. Shah, A. H. Jaikar, S. Bae, and S.-Y. Noh: “**Improve Performance and Throughput of VMs for Scientific Workloads in a Cloud Environment**”. In *Proc. PlatCon*. 2016, pp. 1–6. DOI: 10.1109/PlatCon.2016.7456802
26. S. K. Tesfatsion, E. Wadbro, and J. Tordsson: “**Autonomic Resource Management for Optimized Power and Performance in Multi-Tenant Clouds**”. In *Proc. ICAC*. 2016, pp. 85–94. DOI: 10.1109/ICAC.2016.32
27. F. Pascual and K. Rządca: “**Partition with Side Effects**”. In *Proc. HiPC*. IEEE, 2015, pp. 295–304. DOI: 10.1109/HiPC.2015.52
28. J. von Kistowski, J. Beckett, K.-D. Lange, H. Block, J. A. Arnold, and S. Kounev: “**Energy Efficiency of Hierarchical Server Load Distribution Strategies**”. In *Proc. MASCOTS*. IEEE, 2015, pp. 75–84. DOI: 10.1109/MASCOTS.2015.11

Total: 13

[3] T. Kalibera, L. Bulej, and P. Tůma: “**Automated Detection of Performance Regressions: The Mono Experience**”. In *Proc. 20th IEEE Intl. Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*. IEEE Computer Society, 2005, pp. 183–190. DOI: 10.1109/MASCOT.2005.18. WOS, SCOPUS, CORE 2008 A, Acceptance 46 of 151 (30.5%)

29. A. B. de Oliveira, S. Fischmeister, A. Diwan, M. Hauswirth, and P. F. Sweeney: “**Perphecy: Performance Regression Test Selection Made Simple but Effective**”. In *Proc. ICST*. 2017, pp. 103–113. DOI: 10.1109/ICST.2017.17
30. S. Mostafa, X. Wang, and T. Xie: “**PerfRanker: Prioritization of Performance Regression Tests for Collection-Intensive Software**”. In *Proc. ISSTA*. ACM, 2017, pp. 23–34. DOI: 10.1145/3092703.3092725
31. A. Brunnert and H. Krcmar: “**Continuous Performance Evaluation and Capacity Planning Using Resource Profiles for Enterprise Applications**”. In *Journal of Systems and Software* 123 (2017), pp. 239–262. DOI: 10.1016/j.jss.2015.08.030. WOS, SCOPUS
32. Q. Luo, D. Poshyvanyk, and M. Grechanik: “**Mining Performance Regression Inducing Code Changes in Evolving Software**”. In *Proc. MSR*. ACM, 2016, pp. 25–36. DOI: 10.1145/2901739.2901765. SCOPUS
33. J. Zhou and S. Li: “**Distance Based Root Cause Analysis and Change Impact Analysis of Performance Regressions**”. In *Mathematical Problems in Engineering* 2015 (2015), p. 9. DOI: 10.1155/2015/690829. WOS, SCOPUS
34. Z. M. Jiang and A. E. Hassan: “**A Survey on Load Testing of Large-Scale Software Systems**”. In *IEEE Transactions on Software Engineering* PP.99 (2015), p. 32. DOI: 10.1109/TSE.2015.2445340. SCOPUS
35. P. Huang, X. Ma, D. Shen, and Y. Zhou: “**Performance Regression Testing Target Prioritization via Performance Risk Analysis**”. In *Proc. ICSE*. ACM, 2014, pp. 60–71. DOI: 10.1145/2568225.2568232
36. P. Lengauer, V. Bitto, F. Angerer, P. Grünbacher, and H. Mössenböck: “**Where Has All My Memory Gone? Determining Memory Characteristics of Product Variants Using Virtual-Machine-Level Monitoring**”. In *Proc. VaMoS*. ACM, 2013, 13:1–13:8. DOI: 10.1145/2556624.2556628. SCOPUS
37. J. Vitek and T. Kalibera: “**R3: Repeatability, Reproducibility and Rigor**”. In *ACM SIGPLAN Notices* 47 (4a 2012), pp. 30–36. DOI: 10.1145/2442776.2442781
38. J. Vitek and T. Kalibera: “**Repeatability, Reproducibility, and Rigor in Systems Research**”. In *Proc. EMSOFT*. ACM, 2011, pp. 33–38. DOI: 10.1145/2038642.2038650
39. K. Foo, Z. M. Jiang, B. Adams, A. Hassan, Y. Zou, and P. Flora: “**Mining Performance Regression Testing Repositories for Automated Performance Analysis**”. In *Proc. QSIC*. 2010, pp. 32–41. DOI: 10.1109/QSIC.2010.35
40. N. Roy and D. C. Schmidt: “**Model-Driven Performance Evaluation of Web Application Portals**”. In *Model-Driven Domain Analysis and Software Development: Architectures and Functions*. 2010, pp. 407–437. DOI: 10.4018/978-1-61692-874-2.ch019. SCOPUS

Total: 12

- [4] T. Kalibera, L. Bulej, and P. Tůma: “**Benchmark Precision and Random Initial State**”. In *Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*. SCS, 2005, pp. 853–862. ISBN: 978-1-62276-350-4. SCOPUS
41. A. Maricq, D. Duplyakin, I. Jimenez, C. Maltzahn, R. Stutsman, and R. Ricci: “**Taming Performance Variability**”. In *13th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*. USENIX Association, 2018, pp. 409–425. ISBN: 978-1-931971-47-8
 42. B. R. Bruce, J. Petke, M. Harman, and E. T. Barr: “**Approximate Oracles and Synergy in Software Energy Search Spaces**”. In *IEEE Transactions on Software Engineering* (2018), pp. 1–1. DOI: 10.1109/TSE.2018.2827066. WOS, SCOPUS
 43. E. Barrett, C. F. Bolz-Tereick, R. Killick, S. Mount, and L. Tratt: “**Virtual Machine Warmup Blows Hot and Cold**”. In *Proc. ACM Program. Lang.* 1 (OOPSLA 2017), 52:1–52:27. DOI: 10.1145/3133876
 44. D. Beyer, S. Löwe, and P. Wendler: “**Reliable Benchmarking: Requirements and Solutions**”. In *International Journal on Software Tools for Technology Transfer* (2017), pp. 1–29. DOI: 10.1007/s10009-017-0469-y
 45. A. Brunnert and H. Krcmar: “**Continuous Performance Evaluation and Capacity Planning Using Resource Profiles for Enterprise Applications**”. In *Journal of Systems and Software* 123 (2017), pp. 239–262. DOI: 10.1016/j.jss.2015.08.030. WOS, SCOPUS

46. A. B. de Oliveira, J.-C. Petkovich, and S. Fischmeister: “**How Much Does Memory Layout Impact Performance? A Wide Study**”. In *Proc. REPRODUCE 2014*. 2014, pp. 23–28
47. A. B. de Oliveira, S. Fischmeister, A. Diwan, and M. Hauswirth: “**Why You Should Care About Quantile Regression**”. In *ACM SIGPLAN Notices* 48.4 (2013), pp. 207–217. ISSN: 0362-1340. WOS, SCOPUS
48. J. Larres, A. Potanin, and Y. Hirose: “**A Study of Performance Variations in the Mozilla Firefox Web Browser**”. In *Proc. ACSC 2013*. Australian Computer Society, Inc., 2013, pp. 3–12. ISBN: 978-1-921770-20-3. SCOPUS
49. J. Vanhie-Van Gerwen, S. Bouckaert, I. Moerman, and P. Demeester: “**Benchmarking for Wireless Sensor Networks**”. In *Proc. SENSORCOMM 2011*. 2011, pp. 134–139. ISBN: 978-1-61208-144-1. SCOPUS
50. N. Roy and D. C. Schmidt: “**Model-Driven Performance Evaluation of Web Application Portals**”. In *Model-Driven Domain Analysis and Software Development: Architectures and Functions*. 2010, pp. 407–437. DOI: 10.4018/978-1-61692-874-2.ch019. SCOPUS
51. T. Mytkowicz, A. Diwan, M. Hauswirth, and P. F. Sweeney: “**Producing Wrong Data Without Doing Anything Obviously Wrong!**” In *ACM SIGPLAN Notices* 44.3 (2009), pp. 265–276. DOI: 10.1145/1508284.1508275. WOS, SCOPUS
52. T. Mytkowicz, A. Diwan, M. Hauswirth, and P. Sweeney: “**We Have It Easy, But Do We Have It Right?**” In *Proc. IPDPS 2008*. 2008, pp. 1–7. DOI: 10.1109/IPDPS.2008.4536408. WOS, SCOPUS

Total: 12

[5] L. Bulej, T. Bureš, T. Coupaye, M. Děcký, P. Ježek, P. Parížek, F. Plášil, T. Poch, N. Rivierre, O. Šerý, and P. Tůma: “**CoCoME in Fractal**”. In *The Common Component Modeling Example*. LNCS 5153. Springer, 2008, pp. 357–387. DOI: 10.1007/978-3-540-85289-6_14. WOS, SCOPUS

53. A. Lahcen, D. Parigot, and S. Mouline: “**Data-Dependency Formalism for Developing Peer-To-Peer Applications**”. In *Computing and Informatics* 36.2 (2017), pp. 353–385. DOI: 10.4149/cai2017_2_353. SCOPUS
54. J. A. Fernández-Madrugal, L. Llopis, A. Cruz-Martín, C. Galindo, and J. González-Jiménez: “**H: A Component-Based Specification Language for Heterogeneous Applications**”. In *Computer Standards & Interfaces* 35.1 (2013), pp. 30–49. DOI: 10.1016/j.csi.2012.03.003. SCOPUS
55. H. Aris and S. S. Salim: “**Properties for Component Model: The Definition Perspective**”. In *Computing and Informatics* 30.5 (2012), pp. 987–1010. ISSN: 1335-9150. WOS
56. C. Canal and A. Cansado: “**Component Reconfiguration in Presence of Mismatch**”. In *Informatica* 35.1 (2011). SCOPUS
57. A. Both, W. Zimmermann, and R. Franke: “**Model Checking of Component Protocol Conformance - Optimizations by Reducing False Negatives**”. In *Electronic Notes in Theoretical Computer Science*. Proc. FACS 2009 263 (2010), pp. 67–94. DOI: 10.1016/j.entcs.2010.05.005
58. A. Cansado and C. Canal: “**On the Reconfiguration of Components in Presence of Mismatches**”. In *Proc. WASELF 2009*. Sistedes, 2009. ISSN: 1988–3455
59. A. Plšek and J. Adámek: “**Carmen: Software Component Model Checker**”. In *Proc. QOSA 2008*. LNCS 5281. Springer, 2008, pp. 71–85. DOI: 10.1007/978-3-540-87879-7_5. WOS
60. P. Vařeková, B. Zimmerová, P. Moravec, and I. Černá: “**Formal Verification of Systems with an Unlimited Number of Components**”. In *IET Software* 2.6 (2008), pp. 532–546. DOI: 10.1049/iet-sen:20080009. WOS
61. P. Vařeková and I. Černá: “**Model Checking of Control-User Component-Based Parametrised Systems**”. In *Proc. CBSE 2008*. LNCS 5282. Springer, 2008, pp. 146–162. DOI: 10.1007/978-3-540-87891-9_10. WOS, SCOPUS
62. P. Parížek and O. Lhoták: “**Model Checking of Concurrent Programs with Static Analysis of Field Accesses**”. In *Science of Computer Programming* 98, Part 4 (2015), pp. 735–763. DOI: 10.1016/j.scico.2014.10.008. SCOPUS
63. P. Parížek and O. Lhoták: “**Identifying Future Field Accesses in Exhaustive State Space Traversal**”. In *Proc. ASE 2011*. IEEE Computer Society, 2011, pp. 93–102. DOI: 10.1109/ASE.2011.6100154. SCOPUS

- [6] A. Sarimbekov, A. Podzimek, L. Bulej, Y. Zheng, N. Ricci, and W. Binder: “**Characteristics of Dynamic JVM Languages**”. In *Proc. 7th ACM Workshop on Virtual Machines and Intermediate Languages (VMIL)*. ACM, 2013, pp. 11–20. DOI: 10.1145/2542142.2542144. SCOPUS
64. S. Gaikwad, A. Nisbet, and M. Luján: “**Performance Analysis for Languages Hosted on the Truffle Framework**”. In *Proceedings of the 15th International Conference on Managed Languages & Runtimes*. ACM, 2018, 5:1–5:12. DOI: 10.1145/3237009.3237019
65. B. Ilbeyi, C. F. Bolz-Tereick, and C. Batten: “**Cross-Layer Workload Characterization of Meta-Tracing JIT VMs**”. In *2017 IEEE International Symposium on Workload Characterization (IISWC)*. 2017, pp. 97–107. DOI: 10.1109/IISWC.2017.8167760
66. B.-S. Yang, J.-Y. Kim, and S.-M. Moon: “**Exceptionization: A Java VM Optimization for Non-Java Languages**”. In *ACM Trans. Archit. Code Optim.* 14.1 (2017), 5:1–5:25. DOI: 10.1145/3046681
67. M. Kmjetin and N. Četić: “**Scripting Framework for Graphical Configuration Tool for Embedded Devices**”. In *Proc. 24th Telecommunications Forum (TELFOR)*. IEEE, 2016, pp. 1–4. DOI: 10.1109/TELFOR.2016.7818914
68. E. Barrett, C. F. Bolz, L. Diekmann, and L. Tratt: “**Fine-Grained Language Composition: A Case Study**”. In *Proc. ECOOP 2016*. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2016, 3:1–3:27. DOI: <http://dx.doi.org/10.4230/LIPIcs.ECOOP.2016.3>
69. S. Chidambaram, S. Saraswati, R. Ramachandra, J. B. Huttanagoudar, N. Hema, and R. Roopalakshmi: “**JVM Characterization Framework for Workload Generated as per Machine Learning Benchmark and Spark Framework**”. In *Proc. RTEICT*. 2016, pp. 1598–1602. DOI: 10.1109/RTEICT.2016.7808102
70. N. Hema, K. G. Srinivasa, S. Chidambaram, S. Saraswat, S. Saraswati, R. Ramachandra, and J. B. Huttanagoudar: “**Performance Analysis of Java Virtual Machine for Machine Learning Workloads Using Apache Spark**”. In *Proceedings of the International Conference on Informatics and Analytics*. ACM, 2016, 125:1–125:7. DOI: 10.1145/2980258.2982117
71. S. Marr, B. Daloz, and H. Mössenböck: “**Cross-Language Compiler Benchmarking: Are We Fast Yet?**” In *Proceedings of the 12th Symposium on Dynamic Languages*. ACM, 2016, pp. 120–131. DOI: 10.1145/2989225.2989232
72. S. Xu, D. Bremner, and D. Heidinga: “**Mining Method Handle Graphs for Efficient Dynamic JVM Languages**”. In *Proc. PPPJ 2015*. ACM, 2015, pp. 159–169. DOI: 10.1145/2807426.2807440
73. E. Barrett, C. F. Bolz, and L. Tratt: “**Approaches to Interpreter Composition**”. In *Computer Languages, Systems & Structures* 44, Part C (2015), pp. 199–217. DOI: 10.1016/j.cl.2015.03.001. WOS, SCOPUS

Total: 10

- [7] A. Buble, L. Bulej, and P. Tůma: “**CORBA Benchmarking: A Course with Hidden Obstacles**”. In *Proc. 17th International Parallel and Distributed Processing Symposium*. 2003, pp. 1–6. DOI: 10.1109/IPDPS.2003.1213501. SCOPUS, Workshop at CORE 2008 A
74. E. Bakshy and E. Frachtenberg: “**Design and Analysis of Benchmarking Experiments for Distributed Internet Services**”. In *Proc. WWW 2015*. ACM, 2015, pp. 108–118. DOI: 10.1145/2736277.2741082. SCOPUS
75. A. Hüick, J. Willkomm, and C. Bischof: “**Source Transformation for the Optimized Utilization of the Matlab Runtime System for Automatic Differentiation**”. In *Proc. CE 2014*. Lecture Notes in Computational Science and Engineering 105. Springer, 2015, pp. 115–131. DOI: 10.1007/978-3-319-22997-3_7. SCOPUS
76. A. Iosup, M. Capotă, T. Hegeman, Y. Guo, W. L. Ngai, A. L. Varbanescu, and M. Verstraaten: “**Towards Benchmarking IaaS and PaaS Clouds for Graph Analytics**”. In *Proc. 5th International Workshop on Big Data Benchmarking (WBDB)*. LNCS 8991. Springer, 2014, pp. 109–131. DOI: 10.1007/978-3-319-20233-4_11. WOS, SCOPUS
77. A. Iosup, R. Prodan, and D. Epema: “**IaaS Cloud Benchmarking: Approaches, Challenges, and Experience**”. In *Cloud Computing for Data-Intensive Applications*. Springer, 2014, pp. 83–104. DOI: 10.1007/978-1-4939-1905-5_4

78. J. Xu, H. Zhang, and Q. Li: “**An Execution Tracing Tool for Multi-Tier Web Applications**”. In *Proc. 2nd International Conference on Computer Science and Information Engineering (CSIE)*. Communications in Computer and Information Science 153. Springer, 2011, pp. 244–250. DOI: 10.1007/978-3-642-21411-0_40. WOS, SCOPUS
79. L. Andrey, O. Festor, A. Lahmadi, A. Pras, and J. Schönwälder: “**Survey of SNMP Performance Analysis Studies**”. In *International Journal of Network Management* 19.6 (2009), pp. 527–548. DOI: 10.1002/nem.729. WOS, SCOPUS
80. C. Esposito, S. Russo, and D. Di Crescenzo: “**Performance Assessment of OMG Compliant Data Distribution Middleware**”. In *Proc. IPDPS 2008*. 2008, pp. 1–8. DOI: 10.1109/IPDPS.2008.4536566. WOS, SCOPUS
81. T. Parsons, A. Mos, M. Trofin, T. Gschwind, and J. Murphy: “**Extracting Interactions in Component-Based Systems**”. In *IEEE Transactions on Software Engineering* 34.6 (2008), pp. 783–799. DOI: 10.1109/TSE.2008.67. WOS, SCOPUS
82. T. Parsons, A. Mos, and J. Murphy: “**Non-Intrusive End-to-End Runtime Path Tracing for J2EE Systems**”. In *IEE Proceedings - Software* 153.4 (2006), p. 149. DOI: 10.1049/ip-sen:20050069. WOS, SCOPUS
83. C. Demarey, G. Harbonnier, R. Rouvoy, and P. Merle: “**Benchmarking the Round-Trip Latency of Various Java-Based Middleware Platforms**”. In *Studia Informatica Universalis* 4.1 (2005), pp. 7–24

Total: 10

[8] L. Bulej, T. Bureš, J. Kezníkl, A. Koubková, A. Podzimek, and P. Tůma: “**Capturing Performance Assumptions Using Stochastic Performance Logic**”. In *Proc. 3rd ACM/SPEC International Conference on Performance Engineering (ICPE)*. ACM, 2012, pp. 311–322. DOI: 10.1145/2188286.2188345. SCOPUS, Acceptance 33 of 66 (50.0%)

84. A. Danciu and H. Krcmar: “**To What Extent Does Performance Awareness Support Developers in Fixing Performance Bugs?**” In *Proc. 15th European Performance Engineering Workshop (EPEW)*. Springer, 2018. DOI: 10.1007/978-3-030-02227-3_2
85. M. Brünink and D. S. Rosenblum: “**Mining Performance Specifications**”. In *Proc. 24th ACM SIGSOFT Intl. Symp. on Foundations of Software Engineering*. ACM, 2016, pp. 39–49. DOI: 10.1145/2950290.2950314. SCOPUS
86. K. Triantafyllidis, W. Aslam, E. Bondarev, J. J. Lukkien, and P. H. N. de With: “**ProMARTES: Accurate Network and Computation Delay Prediction for Component-Based Distributed Systems**”. In *Journal of Systems and Software* 117 (2016), pp. 450–470. DOI: 10.1016/j.jss.2016.03.068. WOS, SCOPUS
87. A. Danciu, A. Chrusciel, A. Brunnert, and H. Krcmar: “**Performance Awareness in Java EE Development Environments**”. In *Proc. EPEW 2015*. LNCS 9272. Springer, 2015, pp. 146–160. DOI: 10.1007/978-3-319-23267-6_10. SCOPUS
88. C. Heger, J. Happe, and R. Farahbod: “**Automated Root Cause Isolation of Performance Regressions During Software Development**”. In *Proc. ICPE*. ACM, 2013, pp. 27–38. DOI: 10.1145/2479871.2479879. SCOPUS
89. M. Hölzl, N. Koch, M. Puviani, M. Wirsing, and F. Zambonelli: “**The Ensemble Development Life Cycle and Best Practices for Collective Autonomic Systems**”. In *Software Engineering for Collective Autonomic Systems*. LNCS 8998. Springer, 2015, pp. 325–354. DOI: 10.1007/978-3-319-16310-9_9. SCOPUS
90. M. Babka, T. Balyo, and J. Kezníkl: “**Solving SMT Problems with a Costly Decision Procedure by Finding Minimum Satisfying Assignments of Boolean Formulas**”. In *Proc. SERA 2014*. Studies in Computational Intelligence 496. Springer, 2014, pp. 231–246. DOI: 10.1007/978-3-319-00948-3_15. SCOPUS
91. T. Martinec, L. Marek, A. Steinhäuser, P. Tůma, Q. Noorshams, A. Rentschler, and R. Reussner: “**Constructing Performance Model of JMS Middleware Platform**”. In *Proc. ICPE 2014*. ACM, 2014, pp. 123–134. DOI: 10.1145/2568088.2568096. SCOPUS

Total: 8

[9] P. Stefan, V. Horký, L. Bulej, and P. Tůma: “**Unit Testing Performance in Java Projects: Are We There Yet?**”. In *Proc. 8th ACM/SPEC International Conference on Performance Engineering (ICPE)*. ACM, 2017, pp. 401–412. DOI: 10.1145/3030207.3030226. Acceptance 24 of 65 (36.9%)

92. O. Karlsson and P. Haller: “**Extending Scala with Records: Design, Implementation, and Evaluation**”. In *Proc. 9th ACM SIGPLAN International Symposium on Scala*. ACM, 2018, pp. 72–82. DOI: 10.1145/3241653.3241661
93. C. Laaber and P. Leitner: “**An Evaluation of Open-Source Software Microbenchmark Suites for Continuous Performance Assessment**”. In *Proceedings of the 15th International Conference on Mining Software Repositories*. ACM, 2018, pp. 119–130. DOI: 10.1145/3196398.3196407
94. H. Gall, C. Alexandru, A. Ciurumelea, G. Grano, C. Laaber, S. Panichella, S. Proksch, G. Schermann, C. Vassallo, and J. Zhao: “**Data-Driven Decisions and Actions in Today’s Software Development**”. In *The Essence of Software Engineering*. Springer, 2018, pp. 137–168. DOI: 10.1007/978-3-319-73897-0_9
95. T. Field, R. Chatley, and D. Wei: “**Software Performance Testing in Virtual Time**”. In *Companion Proc. ICPE 2018*. ACM, 2018, pp. 173–174. DOI: 10.1145/3185768.3186409
96. D. G. Reichelt and S. Kühne: “**Better Early Than Never: Performance Test Acceleration by Regression Test Selection**”. In *Companion Proc. ICPE 2018*. ACM, 2018, pp. 127–130. DOI: 10.1145/3185768.3186289
97. D. G. Reichelt and S. Kühne: “**How to Detect Performance Changes in Software History: Performance Analysis of Software System Versions**”. In *Companion Proc. ICPE 2018*. ACM, 2018, pp. 183–188. DOI: 10.1145/3185768.3186404

Total: 6

[10] L. Bulej, T. Bureš, V. Horký, J. Kotrč, L. Marek, T. Trojánek, and P. Tůma: “**Unit Testing Performance with Stochastic Performance Logic**”. In *Automated Software Engineering 24.1* (2017), pp. 139–187. DOI: 10.1007/s10515-015-0188-0. WOS 2015 IF 1.312, SCOPUS 2015 SNIP 2.647

98. C. Laaber and P. Leitner: “**An Evaluation of Open-Source Software Microbenchmark Suites for Continuous Performance Assessment**”. In *Proceedings of the 15th International Conference on Mining Software Repositories*. ACM, 2018, pp. 119–130. DOI: 10.1145/3196398.3196407
99. H. Gall, C. Alexandru, A. Ciurumelea, G. Grano, C. Laaber, S. Panichella, S. Proksch, G. Schermann, C. Vassallo, and J. Zhao: “**Data-Driven Decisions and Actions in Today’s Software Development**”. In *The Essence of Software Engineering*. Springer, 2018, pp. 137–168. DOI: 10.1007/978-3-319-73897-0_9
100. D. G. Reichelt and S. Kühne: “**How to Detect Performance Changes in Software History: Performance Analysis of Software System Versions**”. In *Companion Proc. ICPE 2018*. ACM, 2018, pp. 183–188. DOI: 10.1145/3185768.3186404
101. P. Hnětynka, P. Kubát, R. Al-Ali, I. Gerostathopoulos, and D. Khalyeyev: “**Guaranteed Latency Applications in Edge-Cloud Environment**”. In *Companion Proc. ECSA 2018*. ACM, 2018, 43:1–43:4. DOI: 10.1145/3241403.3241448

[11] S. Becker, L. Bulej, T. Bureš, P. Hnětynka, L. Kapová, J. Kofroň, H. Kozirolek, J. Kraft, R. Mirandola, J. Stammel, G. Tamburrelli, and M. Trifu: “**Q-ImPrESS Project Deliverable D2.1: Service Architecture Meta Model (SAMM)**”. tech. rep. D2.1. Q-ImPrESS Consortium, 2008, p. 109

102. H. Kozirolek, B. Schlich, S. Becker, and M. Hauck: “**Performance and Reliability Prediction for Evolving Service-Oriented Software Systems**”. In *Empirical Software Engineering 18.4* (2012), pp. 746–790. DOI: 10.1007/s10664-012-9213-0. WOS, SCOPUS
103. B. Klatt, C. Rathfelder, and S. Kounev: “**Integration of Event-Based Communication in the Palladio Software Quality Prediction Framework**”. In *Proc. CompArch 2011*. ACM, 2011, pp. 43–52. DOI: 10.1145/2000259.2000268. SCOPUS
104. H. Kozirolek, B. Schlich, C. Bilich, R. Weiss, S. Becker, K. Krogmann, M. Trifu, R. Mirandola, and A. Kozirolek: “**An Industrial Case Study on Quality Impact Prediction for Evolving Service-Oriented Software**”. In *Proceedings of the 33rd International Conference on Software Engineering*. ACM, 2011, pp. 776–785. DOI: 10.1145/1985793.1985902. WOS, SCOPUS
105. J. Stammel and M. Trifu: “**Tool-Supported Estimation of Software Evolution Effort in Service Oriented Systems**”. In *Proc. CSMR 2011*. CEUR Workshop Proceedings, 2011, pp. 56–63. ISSN: 1613-0073, SCOPUS

- [12] L. Bulej and T. Bureš: “**Using Connectors for Deployment of Heterogeneous Applications in the Context of OMG D&C Specification**”. In *Interoperability of Enterprise Software and Applications*. Springer, 2006, pp. 349–360. DOI: 10.1007/1-84628-152-0_31. WOS, Acceptance 35 of 85 (41.2%)
106. A. D. Salov, H. S. Park, S. Han, and D. Lee: “**An Effective Method of Sharing Heterogeneous Components of OPROS and RTM**”. in *Journal of Electrical Engineering and Technology* 9.2 (2014), pp. 755–761. DOI: 10.5370/JEET.2014.9.2.755. WOS
107. P. Martinez, C. Cuevas, and J. Drake: “**RT-D&C: Deployment Specification of Real-Time Component-Based Applications**”. In *Proc. SEAA 2010*. 2010, pp. 147–155. DOI: 10.1109/SEAA.2010.22
108. A. Heydarnoori: “**Deploying Component-Based Applications: Tools and Techniques**”. In *Software Engineering Research, Management and Applications*. Studies in Computational Intelligence 150. Springer, 2008, pp. 29–42. DOI: 10.1007/978-3-540-70561-1_3. WOS
109. A. Heydarnoori, F. Mavaddat, and F. Arbab: “**Towards an Automated Deployment Planner for Composition of Web Services as Software Components**”. In *Electronic Notes in Theoretical Computer Science*. Proceedings of the International Workshop on Formal Aspects of Component Software (FACS 2005) 160 (2006), pp. 239–253. DOI: 10.1016/j.entcs.2006.05.026
- [13] L. Marek, S. Kell, Y. Zheng, L. Bulej, W. Binder, P. Tůma, D. Ansaloni, A. Sarimbekov, and A. Sewe: “**ShadowVM: Robust and Comprehensive Dynamic Program Analysis for the Java Platform**”. In *ACM SIGPLAN Notices* 49.3 (2014), pp. 105–114. DOI: 10.1145/2637365.2517219. WOS 2014 IF 0.657, SCOPUS 2014 SNIP 0.790. Also in: “**ShadowVM: Robust and Comprehensive Dynamic Program Analysis for the Java Platform**”. In *Proc. 12th International Conference on Generative Programming: Concepts & Experiences (GPCE)*. ACM, 2013, pp. 105–114. DOI: 10.1145/2517208.2517219. SCOPUS, CORE 2013 B, Acceptance 20 of 59 (33.9%)
110. M. J. Steindorfer and J. J. Vinju: “**Performance Modeling of Maximal Sharing**”. In *Proc. ICPE*. ACM, 2016, pp. 135–146. DOI: 10.1145/2851553.2851566
111. D. Liang, R. Chen, and H. Y. Sun: “**DroidMonitor: A High-Level Programming Model for Dynamic API Monitoring on Android**”. In *Proc. NSCE*. CRC Press, 2014, pp. 93–96. DOI: 10.1201/b18660-22
112. A. Rosà and W. Binder: “**Optimizing Type-Specific Instrumentation on the JVM with Reflective Supertype Information**”. In *Journal of Visual Languages & Computing* 49 (2018), pp. 29–45. DOI: 10.1016/j.jvlc.2018.10.007
113. E. Rosales, A. Rosà, and W. Binder: “**Tgp: A Task-Granularity Profiler for the Java Virtual Machine**”. In *Proc. APSEC*. 2017, pp. 570–575. DOI: 10.1109/APSEC.2017.67
- [14] P. Libič, L. Bulej, V. Horký, and P. Tůma: “**On the Limits of Modeling Generational Garbage Collector Performance**”. In *Proc. 5th ACM/SPEC International Conference on Performance Engineering (ICPE)*. ACM, 2014, pp. 15–26. DOI: 10.1145/2568088.2568097. SCOPUS, Acceptance 14 of 56 (25%), Best Research Paper Award
114. A. O. Portillo-Dominguez: “**Towards an Efficient Benchmark Generation Engine for Garbage Collection**”. In *Companion Proc. ICPE*. ACM, 2018, pp. 9–12. DOI: 10.1145/3185768.3186303
115. X. Ren and Y. Zhangxu: “**Generational Garbage Collection Algorithm Based on Lifespan Prediction**”. In *Proc. FiCloud Workshops*. 2016, pp. 183–187. DOI: 10.1109/W-FiCloud.2016.47
116. J. G. Son, J.-W. Kang, J.-H. An, H.-J. Ahn, H.-J. Chun, and J.-G. Kim: “**Parallel Job Processing Technique for Real-Time Big-Data Processing Framework**”. In *Proc. RACS*. ACM, 2016, pp. 226–229. DOI: 10.1145/2987386.2987429
- [15] S. M. Blackburn, A. Diwan, M. Hauswirth, P. F. Sweeney, J. N. Amaral, V. Babka, W. Binder, T. Brecht, L. Bulej, L. Eeckhout, S. Fischmeister, D. Frampton, R. Garner, A. Georges, L. J. Hendren, M. Hind, A. L. Hosking, R. Jones, T. Kalibera, P. Moret, N. Nystrom, V. Pankratius, and P. Tůma: “**Can You Trust Your Experimental Results?**” Tech. rep. 1. Evaluate Collaboratory, 2012, p. 8
117. D. G. Feitelson: “**From Repeatability to Reproducibility and Corroboration**”. In *SIGOPS Oper. Syst. Rev.* 49.1 (2015), pp. 3–11. DOI: 10.1145/2723872.2723875. SCOPUS
118. T. Hoefler and R. Belli: “**Scientific Benchmarking of Parallel Computing Systems: Twelve Ways to Tell the Masses When Reporting Performance Results**”. In *Proc. SC 2015*. ACM, 2015, 73:1–73:12. DOI: 10.1145/2807591.2807644

119. A. S. Harji, P. A. Buhr, and T. Brecht: **“Our Troubles with Linux Kernel Upgrades and Why You Should Care”**. In *SIGOPS Oper. Syst. Rev.* 47.2 (2013), pp. 66–72. DOI: 10.1145/2506164.2506175
- [16] L. Bulej, T. Bureš, V. Horký, J. Keznikl, and P. Tůma: **“Performance Awareness in Component Systems (Vision Paper)”**. In *Proc. 36th IEEE Annual Computer Software and Applications Conference Workshops*. 2012, pp. 514–519. DOI: 10.1109/COMPSACW.2012.96. SCOPUS, Workshop at CORE 2013 B
120. D. B. Abeywickrama, J. Combaz, V. Horký, J. Keznikl, J. Kofroň, A. L. Lafuente, M. Loreti, A. Margheri, P. Mayer, V. Monreale, U. Montanari, C. Pincirolì, P. Tůma, A. Vandin, and E. Vassev: **“Tools for Ensemble Design and Runtime”**. In *Software Engineering for Collective Autonomic Systems*. LNCS 8998. Springer, 2015, pp. 429–448. DOI: 10.1007/978-3-319-16310-9_13. SCOPUS
121. M. Hölzl, N. Koch, M. Puviani, M. Wirsing, and F. Zambonelli: **“The Ensemble Development Life Cycle and Best Practices for Collective Autonomic Systems”**. In *Software Engineering for Collective Autonomic Systems*. LNCS 8998. Springer, 2015, pp. 325–354. DOI: 10.1007/978-3-319-16310-9_9. SCOPUS
122. M. Babka, T. Balyo, and J. Keznikl: **“Solving SMT Problems with a Costly Decision Procedure by Finding Minimum Satisfying Assignments of Boolean Formulas”**. In *Proc. SERA 2014*. Studies in Computational Intelligence 496. Springer, 2014, pp. 231–246. DOI: 10.1007/978-3-319-00948-3_15. SCOPUS
- [17] L. Bulej, T. Kalibera, and P. Tůma: **“Regression Benchmarking with Simple Middleware Benchmarks”**. In *Proc. 23rd IEEE International Performance, Computing, and Communications Conference*. IEEE, 2004, pp. 771–776. DOI: 10.1109/PCCC.2004.1395179. WOS, SCOPUS, Workshop at CORE 2008 B
123. A. Hück, J. Willkomm, and C. Bischof: **“Source Transformation for the Optimized Utilization of the Matlab Runtime System for Automatic Differentiation”**. In *Proc. CE 2014*. Lecture Notes in Computational Science and Engineering 105. Springer, 2015, pp. 115–131. DOI: 10.1007/978-3-319-22997-3_7. SCOPUS
124. K. Foo, Z. M. Jiang, B. Adams, A. Hassan, Y. Zou, and P. Flora: **“Mining Performance Regression Testing Repositories for Automated Performance Analysis”**. In *Proc. QSIC*. 2010, pp. 32–41. DOI: 10.1109/QSIC.2010.35
125. M. Procházka, A. Madan, J. Vitek, and W. Liu: **“RTJBench: A Real-Time Java Benchmarking Framework.”** In *Studia Informatica Universalis* 4.1 (2005), pp. 73–82
- [18] T. Kalibera, L. Bulej, and P. Tůma: **“Generic Environment for Full Automation of Benchmarking”**. In *Proc. 1st International Workshop on Software Quality (SOQUA)*. GI, 2004, pp. 125–132. ISBN: 3-88579-387-3
126. M. Procházka, A. Madan, J. Vitek, and W. Liu: **“RTJBench: A Real-Time Java Benchmarking Framework.”** In *Studia Informatica Universalis* 4.1 (2005), pp. 73–82
127. D. Westermann, J. Happe, M. Hauck, and C. Heupel: **“The Performance Cockpit Approach: A Framework For Systematic Performance Evaluations”**. In *Proc. SEAA 2010*. 2010, pp. 31–38. DOI: 10.1109/SEAA.2010.24
128. D. Westermann and J. Happe: **“Towards Performance Prediction of Large Enterprise Applications Based on Systematic Measurements”**. In *Proc. WCOP 2010*. 2010, pp. 71–78
- [19] S. M. Blackburn, A. Diwan, M. Hauswirth, P. F. Sweeney, J. N. Amaral, T. Brecht, L. Bulej, C. Click, L. Eeckhout, S. Fischmeister, D. Frampton, L. J. Hendren, M. Hind, A. L. Hosking, R. E. Jones, T. Kalibera, N. Keynes, N. Nystrom, and A. Zeller: **“The Truth, The Whole Truth, and Nothing But the Truth: A Pragmatic Guide to Assessing Empirical Evaluations”**. In *ACM Transactions on Programming Languages and Systems* 38.4 (2016), 15:1–15:20. DOI: 10.1145/2983574. WOS 2015 IF 1.148, SCOPUS 2015 SNIP 3.141
129. M. Matos: **“Towards Reproducible Evaluation of Large-Scale Distributed Systems”**. In *Proceedings of the 2018 Workshop on Advanced Tools, Programming Languages, and PLatforms for Implementing and Evaluating Algorithms for Distributed Systems*. ACM, 2018, pp. 5–7. DOI: 10.1145/3231104.3231113
130. R. Biddle: **“How Can Our Publication Models Best Serve Our Research? (Panel)”**. In *Proceedings of the 2017 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software*. ACM, 2017, pp. 260–261. DOI: 10.1145/3133850.3144535
- [20] Y. Zheng, S. Kell, L. Bulej, H. Sun, and W. Binder: **“Comprehensive Multi-Platform Dynamic Program Analysis for Java and Android”**. In *IEEE Software* 33.4 (2016), pp. 55–63. DOI: 10.1109/MS.2015.151. WOS 2015 IF 0.820, SCOPUS 2015 SNIP 1.745

131. M. Jaber, Y. Falcone, K. Dak-Al-Bab, J. Abou-Jaoudeh, and M. El-Katerji: “**A High-Level Modeling Language for the Efficient Design, Implementation, and Testing of Android Applications**”. In *International Journal on Software Tools for Technology Transfer* 20.1 (2018), pp. 1–18. DOI: 10.1007/s10009-016-0441-2
132. C.-T. Lin, K.-W. Tang, J.-S. Wang, and G. M. Kapfhammer: “**Empirically Evaluating Greedy-Based Test Suite Reduction Methods at Different Levels of Test Suite Complexity**”. In *Science of Computer Programming* 150 (2017), pp. 1–25. DOI: 10.1016/j.scico.2017.05.004
- [21] L. Marek, Y. Zheng, D. Ansaloni, L. Bulej, A. Sarimbekov, W. Binder, and P. Tůma: “**Introduction to Dynamic Program Analysis with DiSL**”. in *Science of Computer Programming* 98, Part 1 (2015), pp. 100–115. DOI: 10.1016/j.scico.2014.01.003. WOS 2015 IF 0.828, SCOPUS 2015 SNIP 1.380
133. E. Bergen and S. Edlich: “**Post-Debugging in Large Scale Big Data Analytic Systems**”. In *Proc. BTW 2017*. Gesellschaft für Informatik e.V., 2017, pp. 65–72. ISBN: 978-3-88579-660-2
134. F. Marchand de Kerchove, J. Noyé, and M. Südholt: “**Towards Modular Instrumentation of Interpreters in JavaScript**”. In *Proc. MODULARITY FOAL*. ACM, 2015, pp. 64–69. DOI: 10.1145/2735386.2736753
- [22] L. Bulej, T. Bureš, I. Gerostathopoulos, V. Horký, J. Keznikl, L. Marek, M. Tschaikowski, M. Tribastone, and P. Tůma: “**Supporting Performance Awareness in Autonomous Ensembles**”. In *Software Engineering for Collective Autonomic Systems*. LNCS 8998. Springer, 2015, pp. 291–322. DOI: 10.1007/978-3-319-16310-9_8. SCOPUS
135. M. Hölzl, N. Koch, M. Puviani, M. Wirsing, and F. Zambonelli: “**The Ensemble Development Life Cycle and Best Practices for Collective Autonomic Systems**”. In *Software Engineering for Collective Autonomic Systems*. LNCS 8998. Springer, 2015, pp. 325–354. DOI: 10.1007/978-3-319-16310-9_9. SCOPUS
136. P. Mayer, J. Velasco, A. Klarl, R. Hennicker, M. Puviani, F. Tiezzi, R. Pugliese, J. Keznikl, and T. Bureš: “**The Autonomic Cloud**”. In *Software Engineering for Collective Autonomic Systems*. LNCS 8998. Springer, 2015, pp. 495–512. DOI: 10.1007/978-3-319-16310-9_16
- [23] P. Libič, L. Bulej, V. Horký, and P. Tůma: “**Estimating the Impact of Code Additions on Garbage Collection Overhead**”. In *Proc. 12th European Performance Engineering Workshop (EPEW)*. LNCS 9272. Springer, 2015, pp. 130–145. DOI: 10.1007/978-3-319-23267-6_9. WOS, SCOPUS, Acceptance 19 of 39 (48.7%)
137. F. Willnecker and H. Krcmar: “**Optimization of Deployment Topologies for Distributed Enterprise Applications**”. In *Proc. QOSA 2016*. 2016. DOI: 10.1109/QoSA.2016.11
138. F. Willnecker, A. Brunnert, B. Koch-Kemper, and H. Krcmar: “**Full-Stack Performance Model Evaluation Using Probabilistic Garbage Collection Simulation**”. In *Softwaretechnik-Trends* 35.3 (2015), pp. 1–3. ISSN: ISSN 0720-8928
- [24] L. Bulej, T. Bureš, V. Horký, and J. Keznikl: “**Adaptive Deployment in Ad-Hoc Systems Using Emergent Component Ensembles (Vision Paper)**”. In *Proc. 4th ACM/SPEC International Conference on Performance Engineering (ICPE)*. ACM, 2013, pp. 343–346. DOI: 10.1145/2479871.2479922. SCOPUS
139. P. Mayer, J. Velasco, A. Klarl, R. Hennicker, M. Puviani, F. Tiezzi, R. Pugliese, J. Keznikl, and T. Bureš: “**The Autonomic Cloud**”. In *Software Engineering for Collective Autonomic Systems*. LNCS 8998. Springer, 2015, pp. 495–512. DOI: 10.1007/978-3-319-16310-9_16
140. P. Mayer, A. Klarl, R. Hennicker, M. Puviani, F. Tiezzi, R. Pugliese, J. Keznikl, and T. Bureš: “**The Autonomic Cloud: A Vision of Voluntary, Peer-2-Peer Cloud Computing**”. In *Proc. SASO 2013 Workshops*. 2013, pp. 89–94. DOI: 10.1109/SASOW.2013.16
- [25] A. Sarimbekov, A. Sewe, S. Kell, Y. Zheng, W. Binder, L. Bulej, and D. Ansaloni: “**A Comprehensive Toolchain for Workload Characterization Across JVM Languages**”. In *Proc. 11th ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering (PASTE)*. ACM, 2013, pp. 9–16. DOI: 10.1145/2462029.2462033. SCOPUS, CORE 2013 B, Acceptance 7 of 13 (53.8%)
141. S. Xu, D. Bremner, and D. Heidinga: “**Mining Method Handle Graphs for Efficient Dynamic JVM Languages**”. In *Proc. PPPJ 2015*. ACM, 2015, pp. 159–169. DOI: 10.1145/2807426.2807440
142. W. H. Li, D. R. White, and J. Singer: “**JVM-Hosted Languages: They Talk the Talk, but Do They Walk the Walk?**” In *Proc. PPPJ 2013*. ACM, 2013, pp. 101–112. DOI: 10.1145/2500828.2500838

- [26] Y. Zheng, L. Bulej, C. Zhang, S. Kell, D. Ansaloni, and W. Binder: “**Dynamic Optimization of Bytecode Instrumentation**”. In *Proc. 7th ACM Workshop on Virtual Machines and Intermediate Languages (VMIL)*. ACM, 2013, pp. 21–30. DOI: 10.1145/2542142.2542145. SCOPUS
143. P. Arafa, G. M. Tchamgoue, H. Kashif, and S. Fischmeister: “**QDIME: QoS-Aware Dynamic Binary Instrumentation**”. In *Proc. MASCOTS 2017*. 2017, pp. 132–142. DOI: 10.1109/MASCOTS.2017.19
144. C. Xiang, Z. Qi, and W. Binder: “**Flexible and Extensible Runtime Verification for Java (Extended Version)**”. In *International Journal of Software Engineering and Knowledge Engineering* 25 (09n10 2015), pp. 1595–1609. DOI: 10.1142/S0218194015400343. WOS, SCOPUS
- [27] Y. Zheng, L. Bulej, and W. Binder: “**Accurate Profiling in the Presence of Dynamic Compilation**”. In *ACM SIGPLAN Notices* 50.10 (2015), pp. 433–450. DOI: 10.1145/2858965.2814281. WOS 2015 IF 0.488, SCOPUS 2015 SNIP 0.803. Also in: “**Accurate Profiling in the Presence of Dynamic Compilation**”. In *Proc. 30th ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)*. ACM, 2015, pp. 433–450. DOI: 10.1145/2814270.2814281. CORE 2014 A*, Acceptance 53 of 210 (25.2%), Distinguished Paper Award, Evaluated Artifact
145. J. Fu, G. Jin, L. Zhang, and J. Wang: “**CAOS: Combined Analysis with Online Sifting for Dynamic Compilation Systems**”. In *Proc. CF 2016*. ACM, 2016, pp. 110–118. DOI: 10.1145/2903150.2903151. SCOPUS
146. A. Rosà and W. Binder: “**Optimizing Type-Specific Instrumentation on the JVM with Reflective Supertype Information**”. In *Journal of Visual Languages & Computing* 49 (2018), pp. 29–45. DOI: 10.1016/j.jvlc.2018.10.007
- [28] Y. Zheng, L. Bulej, and W. Binder: “**An Empirical Study on Deoptimization in the Graal Compiler**”. In *Proc. 31st European Conference on Object-Oriented Programming (ECOOP)*. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017, 30:1–30:30. DOI: 10.4230/LIPIcs.ECOOP.2017.30. CORE 2017 A, Acceptance 27 of 81 (33.3%)
147. Z. Majo, T. Hartmann, M. Mohler, and T. R. Gross: “**Integrating Profile Caching into the HotSpot Multi-Tier Compilation System**”. In *Proc. 14th Intl. Conf. on Managed Languages and Runtimes (MANLANG)*. ACM, 2017, pp. 105–118. DOI: 10.1145/3132190.3132210
- [29] L. Bulej: “**Performance Testing in Software Development: Getting the Developers on Board (Invited Talk Abstract)**”. In *Companion Proc. 7th ACM/SPEC International Conference on Performance Engineering*. ACM, 2016, pp. 9–9. DOI: 10.1145/2859889.2880448
148. S. Kim, J. Park, K. H. Kim, and J. G. Shon: “**A Test Data Generation for Performance Testing in Massive Data Processing Systems**”. In *Proc. MUE*. Springer, 2018, pp. 207–213. DOI: 10.1007/978-981-13-1328-8_26
- [30] A. Sarimbekov, L. Stadler, L. Bulej, A. Sewe, A. Podzimek, Y. Zheng, and W. Binder: “**Workload Characterization of JVM Languages**”. In *Software: Practice and Experience* 46.8 (2016), pp. 1053–1089. DOI: 10.1002/spe.2337. WOS 2015 IF 0.652, SCOPUS 2015 SNIP 1.311
149. M. R. Azadmanesh, M. L. V. D. Vanter, and M. Hauswirth: “**Language-Independent Information Flow Tracking Engine for Program Comprehension Tools**”. In *Proc. 25th IEEE/ACM Intl. Conf. on Program Comprehension (ICPC)*. 2017, pp. 346–355. DOI: 10.1109/ICPC.2017.5
- [31] H. Sun, Y. Zheng, L. Bulej, W. Binder, and S. Kell: “**Custom Full-Coverage Dynamic Program Analysis for Android (Demo Paper)**”. In *Companion Proc. 2015 ACM SIGPLAN International Conference on Systems, Programming, Languages and Applications: Software for Humanity (SPLASH)*. ACM, 2015, pp. 7–8. DOI: 10.1145/2814189.2814190
150. Y. Cha and W. Pak: “**Protecting Contacts against Privacy Leaks in Smartphones**”. In *PLOS ONE* 13.7 (2018), e0191502. DOI: 10.1371/journal.pone.0191502
- [32] H. Sun, Y. Zheng, L. Bulej, A. Villazón, Z. Qi, P. Tüma, and W. Binder: “**A Programming Model and Framework for Comprehensive Dynamic Analysis on Android**”. In *Proc. 14th International Conference on Modularity (AOSD/MODULARITY)*. ACM, 2015, pp. 133–145. DOI: 10.1145/2724525.2724566. CORE 2014 A (AOSD), Acceptance 11 of 29 (37.9%)
151. D. Kim, S. Kim, and J. Ryou: “**Design and Implementation of User-Level Dynamic Binary Instrumentation on ARM Architecture**”. In *The Journal of Supercomputing* (2016), pp. 1–13. DOI: 10.1007/s11227-016-1777-9

- [33] A. Podzimek, L. Y. Chen, L. Bulej, W. Binder, and P. Tůma: **“Showstopper: The Partial CPU Load Tool (Tool Paper)”**. In *Proc. 22nd IEEE International Symposium on Modelling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS)*. IEEE, 2014, pp. 510–513. DOI: 10.1109/MASCOTS.2014.75. SCOPUS, CORE 2014 A
152. W. Lund and J. Lilius: **“Inactivity Benchmarking”**. In *Proc. HPCS*. 2018, pp. 1028–1033. DOI: 10.1109/HPCS.2018.00161
- [34] D. Ansaloni, S. Kell, Y. Zheng, L. Bulej, W. Binder, and P. Tůma: **“Enabling Modularity and Reuse in Dynamic Program Analysis Tools for the Java Virtual Machine”**. In *Proc. 27th European Conference on Object-Oriented Programming (ECOOP)*. LNCS 7920. Springer, 2013, pp. 352–377. DOI: 10.1007/978-3-642-39038-8_15. WOS, SCOPUS, CORE 2013 A, Acceptance 29 of 116 (25%)
153. D. Okanović and M. Vidaković: **“Evaluation of Alternative Instrumentation Frameworks”**. In *Proc. SSP 2014*. 2014, pp. 83–90
- [35] P. Libič, P. Tůma, and L. Bulej: **“Issues in Performance Modeling of Applications with Garbage Collection”**. In *Proc. 1st Intl. W. on Quality of Service-Oriented Software Systems (QUASOSS)*. ACM, 2009, pp. 3–10. DOI: 10.1145/1596473.1596477. WOS, SCOPUS, Workshop at CORE A, Acceptance 7 of 9 (77.8%)
154. L. Ribeiro, S. Karnouskos, P. Leitão, J. Barbosa, and M. Hochwallner: **“Performance Assessment Of The Integration Between Industrial Agents And Low-Level Automation Functions”**. In *Proc. INDIN*. 2018, pp. 121–126. DOI: 10.1109/INDIN.2018.8471927
- [36] L. Bulej and T. Bureš: **“A Connector Model Suitable for Automatic Generation of Connectors”**. Tech. rep. 2003/1. Dept. of SW Engineering, Charles University, 2003
155. W. R. Otte, A. Gokhale, D. C. Schmidt, and J. Willemsen: **“Infrastructure for Component-Based DDS Application Development”**. In *ACM SIGPLAN Notices* 47.3 (2012), pp. 53–61. DOI: 10.1145/2189751.2047872. WOS