Automated Benchmarking of Cloud-Hosted DBMS With benchANT

Daniel Seybold[a], Jörg Domaschka[a]

[a]*Ulm University, Institute of Information Resource Management Albert-Einstein-Allee 43, 89077, Ulm, Germany*

Keywords
DBMS, Cloud, Performance, Scalability, Benchmarking-as-a-Service

Driven by the data-intensive applications of Web 2.0, Big Data and Internet of Things, Database Management Systems (DBMSs) and their operation have significantly changed over the last decade. Besides relational DBMSs, manifold NoSQL [1, 2] and NewSQL [3, 2] DBMSs evolved, promising a set of non-functional features that are key requirements for each data-intensive application: high performance, horizontal scalability, elasticity and high-availability [4].

In order to take full advantage of these non-functional features, the operation of DBMSs is moving towards elastic infrastructures such as the cloud. Cloud computing enables scalability and elasticity on the resource level. Therefore, the storage backend of data-intensive applications is commonly implemented by distributed DBMSs operated on cloud resources [5].

Yet, the sheer number of heterogeneous DBMSs, cloud resource offers, and the resulting number of combinations makes the selection and operation of DBMSs a very challenging task [6, 7]. Therefore, supportive evaluation of the non-functional DBMS features are essential. Here, the design and execution that analyses is a complex process that involves detailed domain knowledge of multiple domains [8, 9, 10]. First, the multitude of DBMSs technologies with their respective runtime parameters needs to be considered. Secondly, the tremendous number of resource offers including their volatile characteristics needs to be taken into account. Thirdly, the application-specific workload has to be created by suitable DBMS benchmarks. While supportive DBMS benchmarks only focus on DBMS performance, the evaluation design and execution for advanced non-functional features such as scalability, elasticity and availability is even more challenging [11].

In order to address these challenges, we present the novel Benchmarking-as-a-Service (BaaS) platform *benchANT*[1] that fully automates the benchmarking process of cloud-hosted DBMS. benchANT is a spin-off of the Ulm University and consequently build on our latest research results in cloud and DBMS performance engineering [12]. In particular, our research results define a supportive evaluation methodology consisting of: (i) domain-specific impact factors for designing comprehensive DBMS evaluations; (ii) a set of evaluation principles to ensure significant results. Moreover, our methodology emphasizes reproducible evaluation processes

[1]https://benchant.com/

for the non-functional features performance, scalability, elasticity and availability.

On a technical level, the benchANT platform builds upon our research prototype Mowgli [10], Kaa [13] and King Louie [14]. Mowgli provides a novel DBMS evaluation framework, supporting the design and automated execution of performance and scalability evaluation processes. Mowgli manages cloud resources, DBMS deployment, workload execution and result processing based on evaluation scenarios, which expose configurable domain-specific parameters. The Kaa framework [13] automates the DBMS elasticity evaluation process by enabling DBMS and workload adaptations. The King Louie framework [14] builds upon these features and enables availability evaluations by providing an extensive failure injection framework.

benchANT lifts these research results into an enterprise grade BaaS platform with an easy-to-use benchmark configurator. In its current state, benchANT enables the configuration and automated execution from 7 major RDBMS, NoSQL and NewSQL DBMS[2] with DBMS-specific runtime configurations such as cluster size, replication factor or consistency settings; 4 public cloud providers[3] with over 700 different VM flavours and 1 benchmark[4] with 5 workload configurations. The evaluation results are automatically processed by the benchANT platform and presented in a result dashboard. The result processing uses the raw DBMS benchmark metrics throughput and latency to generate higher-level metrics such as the scalability factor [10] or unified metrics over the dimensions performance, costs and availability [15].

In this talk, we provide an overview how these research results are incorporated into the novel BaaS concept and demonstrate in a live walk-through how benchANT supports practitioners and researchers to address performance challenges such as:

- Which cloud provider and which VM flavour provides the best performance/cost ratio for a 3 node MongoDB cluster?
- Will new DBMS releases always increase the the performance?
- Is there a significant throughput and latency difference between MongoDB, Cassandra and Couchbase for an IoT workload?

References

[1] A. Davoudian, L. Chen, M. Liu, A survey on nosql stores, ACM Comput. Surv. 51 (2018). doi:10.1145/3158661.

[2] S. Mazumdar, D. Seybold, K. Kritikos, Y. Verginadis, A survey on data storage and placement methodologies for cloud-big data ecosystem, Journal of Big Data 6 (2019) 15. doi:10.1186/s40537-019-0178-3.

[3] K. Grolinger, W. A. Higashino, A. Tiwari, M. A. Capretz, Data management in cloud environments: Nosql and newsql data stores, Journal of Cloud Computing: Advances, Systems and Applications 2 (2013) 22. doi:10.1186/2192-113X-2-22.

[4] D. Abadi, R. Agrawal, A. Ailamaki, M. Balazinska, P. A. Bernstein, M. J. Carey, S. Chaudhuri, J. Dean, A. Doan, M. J. Franklin, J. Gehrke, L. M. Haas, A. Y. Halevy, J. M. Hellerstein, Y. E. Ioannidis, H. V. Jagadish, D. Kossmann, S. Madden, S. Mehrotra, T. Milo, J. F. Naughton,

---

[2]MySQL, PostgreSQL, ArangoDB, Apache Cassandra, Couchbase, MongoDB and CockroachDB
[3]AWS, Azure, IONOS and Telekom
[4]Yahoo Cloud Serving Benchmark (YCSB)

R. Ramakrishnan, V. Markl, C. Olston, B. C. Ooi, C. Ré, D. Suciu, M. Stonebraker, T. Walter, J. Widom, The beckman report on database research, Commun. ACM 59 (2016) 92–99. URL: https://doi.org/10.1145/2845915. doi:10.1145/2845915.

[5] D. Abadi, A. Ailamaki, D. Andersen, P. Bailis, M. Balazinska, P. Bernstein, P. Boncz, S. Chaudhuri, A. Cheung, A. Doan, et al., The seattle report on database research, SIGMOD Rec. 48 (2020) 44–53. doi:10.1145/3385658.3385668.

[6] S. Sakr, Cloud-hosted databases: technologies, challenges and opportunities, Cluster Computing 17 (2014) 487–502. doi:10.1007/s10586-013-0290-7.

[7] M. Stonebraker, A. Pavlo, R. Taft, M. L. Brodie, Enterprise database applications and the cloud: A difficult road ahead, in: 2014 IEEE International Conference on Cloud Engineering, IEEE, 2014, pp. 1–6. doi:10.1109/IC2E.2014.97.

[8] D. Seybold, Towards a framework for orchestrated distributed database evaluation in the cloud, in: Proceedings of the 18th Doctoral Symposium of the 18th International Middleware Conference, Middleware '17, ACM, New York, NY, USA, 2017, pp. 13–14. doi:10.1145/3152688.3152693.

[9] J. Domaschka, D. Seybold, Towards understanding the performance of distributed database management systems in volatile environments, in: Symposium on Software Performance, volume 39, Gesellschaft für Informatik, 2019, pp. 11–13. URL: https://pi.informatik. uni-siegen.de/stt/39_4/01_Fachgruppenberichte/SSP2019/SSP2019_Domaschka.pdf.

[10] D. Seybold, M. Keppler, D. Gründler, J. Domaschka, Mowgli: Finding your way in the dbms jungle, in: Proceedings of the 2019 ACM/SPEC International Conference on Performance Engineering, ICPE '19, ACM, New York, NY, USA, 2019, pp. 321–332. doi:10.1145/3297663.3310303.

[11] D. Seybold, J. Domaschka, Is distributed database evaluation cloud-ready?, in: European Conference on Advances in Databases and Information Systems (ADBIS) - New Trends in Databases and Information Systems (Short Papers), Springer International Publishing, Cham, 2017, pp. 100–108. doi:10.1007/978-3-319-67162-8_12.

[12] D. Seybold, An automation-based approach for reproducible evaluations of distributed DBMS on elastic infrastructures, Ph.D. thesis, 2021. URL: https://oparu.uni-ulm.de/xmlui/handle/123456789/37430. doi:10.18725/OPARU-37368.

[13] D. Seybold, S. Volpert, S. Wesner, A. Bauer, N. Herbst, J. Domaschka, Kaa: Evaluating elasticity of cloud-hosted dbms, in: 2019 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), 2019, pp. 54–61. doi:10.1109/CloudCom.2019.00020.

[14] D. Seybold, S. Wesner, J. Domaschka, King louie: Reproducible availability benchmarking of cloud-hosted dbms, in: 35th ACM/SIGAPP Symposium on Applied Computing (SAC '20), March 30-April 3, 2020, Brno, Czech Republic, 2020, pp. 144–153. doi:10.1145/3341105.3373968.

[15] J. Domaschka, S. Volpert, D. Seybold, Hathi: An mcdm-based approach to capacity planning for cloud-hosted dbms, in: 2020 IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC), 2020, pp. 143–154. doi:10.1109/UCC48980.2020.00033.