Data-Centric Benchmarking

Jérôme Darmont
Université de Lyon, Lyon 2, ERIC EA 3083
5 avenue Pierre Mendès-France, F69676 Bron Cedex, France
jerome.darmont@univ-lyon2.fr

INTRODUCTION

In data management, both system designers and users casually resort to performance evaluation. On one hand, designers need to test architectural features and hypotheses regarding the actual (vs. theoretical) behavior of a system, especially in terms of response and scalability. Performance tuning also necessitates accurate performance evaluation. On the other hand, users are also keen on comparing the efficiency of different technologies before selecting a software solution. Thence, performance measurement tools are of premium importance in the data management domain.

Performance evaluation by experimentation on a real system is generally referred to as benchmarking. It consists in performing a series of tests on a given system to estimate its performance in a given setting. Typically, a data-centric benchmark is constituted of two main elements: a data model (conceptual schema and extension) and a workload model (set of read and write operations) to apply on this dataset, with respect to a predefined protocol. Both models may be parameterized. Most benchmarks also include a set of simple or composite performance metrics such as response time, throughput, number of input/output operations, disk or memory usage, etc.

The Transaction Processing Performance Council (TPC), a non-profit organization founded in 1988, plays a preponderant role in data-centric benchmarking. Its mission is to issue standard benchmarks, to verify their correct application by the industry, and to publish performance test results. TPC members include all the major industrial actors from the database field.

The aim of this chapter is to present an overview of the major past and present state-of-the-art data-centric benchmarks. Our review includes the TPC standard benchmarks, but also alternative or more specialized benchmarks. We survey benchmarks from three families: transaction benchmarks aimed at On-Line Transaction Processing (OLTP), decision-support benchmarks aimed at On-Line Analysis Processing (OLAP) and big data benchmarks. Eventually, we discuss the issues, tradeoffs and future trends in data-centric benchmarking.

BACKGROUND

Transaction Processing Benchmarks

The first TPC benchmark for relational, transactional databases, TPC-C (TPC, 2010), has been in use since 1992. TPC-C features a complex business database (a classical customer-order-product-supplier model with nine types of tables bearing various structures and sizes) and a workload of diversely complex transactions that are executed concurrently. The performance metric in TPC-C is transaction throughput. As all TPC benchmarks, TPC-C's only parameter is a scale factor SF that determines data size. TPC-C was complemented in 2007 by TPC-E (TPC, 2015a), which simulates a brokerage firm with the aim of being representative of more modern OLTP systems. In its principles and features, TPC-E is otherwise very similar to TPC-C.

There are few alternatives to TPC-C and TPC-E for relational applications. Yet, some benchmarks fit niches where there is no standard benchmark. For instance, OO7 (Carey et al., 1993) and OCB (Darmont & Schneider, 2000) are object-oriented database benchmarks modeling engineering applications, e.g., computer-aided design or software engineering. However, their complexity makes both these benchmarks hard to understand and implement. Moreover, with objects in databases being more commonly managed in object-relational systems nowadays, object-relational benchmarks such as BUCKY (Carey et al., 1997) and BORD (Lee et al., 2000) now seem more relevant. Such benchmarks focus on queries implying object identifiers, inheritance, joins, class and object references, mul-

tivalued attributes, query unnesting, object methods, and abstract data types. However, typical object navigation is considered already addressed by object-oriented benchmarks and is not taken into account. Moreover, object-relational database benchmarks have not evolved since the early 2000's, whereas object-relational database systems have.

Similarly, XML benchmarks aim at comparing the various XML storage and querying solutions developed since the late nineties. From the early so-called XML application benchmarks that implement a mixed XML database that is either data-oriented (structured data) or document-oriented (in general, random texts built from a dictionary), XBench (Yao et al., 2004) stands out. XBench is indeed the only benchmark proposing a true mixed dataset (i.e., data *and* document-oriented) and helping evaluate all the functionalities offered by XQuery. FlexBench (Vranec & Mlýnková, 2009) also tests a large set of data characteristics and proposes query templates that allow modeling multiple types of applications. Finally, Schmidt et al. (2009) and Zhang et al. (2011) propose benchmarks that are specifically tailored for testing logical XML model-based systems, namely native XML and XML-relational database management systems, respectively.

Decision-Support Benchmarks

TPC-H (TPC, 2014a) has long been the only standard decision-support benchmark. It exploits a classical product-order-supplier database schema, as well as a workload that is constituted of twenty-two SQL-92, parameterized, decision-support queries and two refreshing functions that insert tuples into and delete tuples from the database. Query parameters are randomly instantiated following a uniform law. Three primary metrics describe performance in terms of power, throughput, and a combination of power and throughput.

However, TPC-H's database schema is not a star-like multidimensional schema that is typical in data warehouses. Furthermore, its workload does not include any true OLAP query. TPC-DS (TPC, 2015b) now fills in this gap. Its schema represents the decision-support functions of a retailer under the form of a constellation schema with several fact tables and shared dimensions. TPC-DS' workload is constituted of four classes of queries: reporting queries, ad-hoc decision-support queries, interactive OLAP queries, and extraction queries. SQL-99 query templates help randomly generate a set of about five hundred queries, following non-uniform distributions. TPC-DS features one primary throughput metric that takes both query execution and data warehouse maintenance into account.

Given the primordial importance of data integration in many data-centric (including data warehousing) scenarios, TPC-H was recently complemented by TPC-DI (TPC, 2014b). TPC-DI focuses on Extract, Load and Transform (ETL) processes. Data are first generated in a staging area as if they were extracted from a virtual retail brokerage firm's operational databases. Then, data are transformed through, e.g., type conversions, attribute splits or merges, and error checks. Finally, data are loaded into a warehouse constituted of five fact tables and eight dimension tables. There are two load phases: an initial, so-called historical load, and then incremental updates. Transformations are different in these two phases. TPC-DI's main metric is a combination of throughputs from the historical load and two incremental updates.

There are, again, few decision-support benchmarks out of the TPC, but with TPC-DS having had an eight-year long development, alternative data warehouse benchmarks were proposed. Published by the OLAP council, a now inactive organization founded by OLAP vendors, APB-1 (OLAP Council, 1998) was the first of them and actually predates TPC-DS. APB-1 has been intensively used in the late nineties. However, APB-1 is very simple and rapidly proved limited to evaluate the specificities of various activities and functions. Thus, more elaborate alternatives were proposed, such as DWEB (Darmont et al., 2007), which can be parameterized to generate various ad-hoc synthetic data warehouses and workloads that include typical OLAP queries, and SSB (O'Neil et al., 2009), which is based on TPC-H's database remodeled as a star schema and features a query workload that provides both functional and selectivity coverage.

It is also worth noting that TPC-DS is a complex benchmark. Thence, simpler benchmarks are still in use, especially for testing OLAP scenarios in the cloud. For instance, TPC-H was used to benchmark Hadoop and Pig (Moussa, 2012) and SSB for testing the efficiency of view materialization

in the cloud (Perriot et al., 2014). Niche benchmarks also rely a lot on TPC-H. XWeB (Mahboubi & Darmont, 2010) proposes a unified reference model for XML warehouses and its associate XQuery decision-support workload. RTDW-bench (Jedrzejczak et al., 2012) is designed for testing the ability of a real-time data warehouse to handle a transaction stream without delay, given an arrival rate. Eventually, Bär and Golab (2012) propose a benchmark for stream data warehouses that measures the freshness of materialized views.

Finally, a couple of benchmarks are even more specific (and unrelated from TPC-H), e.g., Spadawan (Lopes Siqueira et al., 2010), which allows performance evaluation of specific, complex operations in spatial data warehouses, and BenchDW (Triplet & Butler, 2013), which targets biological data warehouses and particularly focuses on performance metrics, with twenty-two different metrics such as documentation quality, accuracy and response time.

Big Data Benchmarks

In the timely trend of big data analytics, benchmarking needs are as high as ever to compare alternative systems, including the many NoSQL database systems. In this context of data variety, volume and velocity, adaptability and scalability are premium features that must be evaluated. The TPC promotes two benchmarks for big data. TPC-VMS (TPC, 2013) is actually a benchmarking environment for virtualized databases that allows running TPC-C, TPC-E, TPC-H or TPC-DS on three virtual machines (VMs). Its metric is the minimum value of the selected benchmark's metric on all VMs. TPCx-HS (TPC, 2015c) focuses on Hadoop and MapReduce-based applications. It models a simple application (with no data model directly available) and features, in addition to classical metrics, availability and energy metrics.

Before the TPC could issue its big data benchmarks, and still in parallel, there are many industrial and academic initiatives. MalStone (Open Cloud Consortium, 2009) is a benchmark for assessing data intensive parallel processing. It features MalGen, a synthetic data generator that produces large datasets generated probabilistically following specified distributions. In the same line, HiBench (Huang et al., 2010) is a set of Hadoop programs, ranging from data sorting to clustering, aimed at measuring metrics such as response time, HDFS bandwidth consumption and data access patterns. SWIM (Chen et al., 2013) also measures the performance of Hadoop/MapReduce systems. SWIM contains suites of workloads of thousands of jobs, with complex data, arrival, and computation patterns, and therefore provides workload-specific optimizations. Finally, HcBench (Saletore et al., 2013) models real Hadoop usages in a datacenter. HcBench features various job types and data sizes.

By contrast, YCSB (Cooper et al., 2010) is a framework that focuses on data, and more specifically on performance evaluation of key-value stores. YCSB defines several metrics and workloads to measure system behavior in different situations, or the same system when using different configurations. OLTP-Bench (Curino et al., 2012) is the first true benchmarking framework designed for cloud transactional database systems as a service. OLTP-Bench actually features a set of existing micro-benchmarks (i.e., designed to test one very specific aspect of performance, e.g., ResourceStresser), popular benchmarks (e.g., TPC-C) and real-world applications (e.g., Wikipedia).

Regarding big data analytics, PRIMEBALL (Ferrarons et al., 2013) aims at providing a real-life context to cloud data warehouse benchmarking. Its authors provide the specifications of a fictitious news site hosted in the cloud that is to be managed by the framework under analysis, together with several objective use cases and measures for evaluating system performance. The Big Data Benchmark (Amplab, 2014) goes one step further by actually implementing existing analytical workload models by Pavlo et al. (2009). Its only metric is the response time of such relational queries as scans, aggregations and joins. It can be used for both MapReduce-based systems (such as Shark and Hive) and classical parallel database systems. BigBench (Rabl et al., 2015) is a so-called a specification-based benchmark that is independent from technology. BigBench relies a lot on TPC-DS, borrowing its data model and part of its workload model. The remainder of the workload is adapted from big data use cases issued by the McKinsey Global Institute. Finally, yet other benchmarks, i.e., CloudSuite (Yasin et al., 2014) and DCBench (Jia et al., 2013), feature machine learning and data mining-oriented workload models that mostly run on Hadoop and exploit the Mahout library.

Eventually, BigDataBench (Wang et al., 2014) aims at providing the widest possible scope of big data models and workloads. It includes nineteen benchmarks representing a large variety of data models, workload models and application scenarios from search engines, social networks, e-commerce, multimedia analytics and bioinformatics. Workload models cover OLTP, "cloud OLTP" and OLAP. As BigBench, BigDataBench also allows alternative implementations, e.g., using MapReduce or Spark.

ISSUES AND TRADEOFFS IN DATA PROCESSING BENCHMARKS

Gray (1993) defines four primary criteria to specify a "good" benchmark.

1. Relevance: The benchmark must deal with aspects of performance that appeal to the largest number of potential users.
2. Portability: The benchmark must be reusable to test the performances of different DBMSs.
3. Simplicity: The benchmark must be feasible and must not require too many resources.
4. Scalability: The benchmark must adapt to small or large computer architectures.

In their majority, existing benchmarks aim at comparing the performances of different systems in given experimental conditions. This helps vendors position their products relatively to their competitors', and users achieve strategic and costly software choices based on objective information. These benchmarks invariably present fixed data and workload models. Gray's scalability factor is achieved through a reduced number of parameters that mainly allows varying database size in predetermined proportions. All TPC benchmarks notably feature a single scale factor parameter.

This solution is simple (still according to Gray's criteria), but the relevance of such benchmarks is inevitably reduced to the test cases that are explicitly modeled. For instance, the typical customer-order-product-supplier data model from TPC benchmarks is unsuitable to many application domains. This leads benchmark users to design more or less elaborate variants of standard tools, when they feel these are not generic enough to fulfill particular needs. Such users are generally not confronted to software choices, but to architectural choices or performance optimization tradeoffs within a given system or family of systems. In this context, it is essential to multiply experiments and test cases, and a monolithic benchmark is of reduced relevance.

To enhance the relevance of benchmarks aimed at system designers, three solutions are possible. The first one is to design an ad-hoc benchmark for a particular application, e.g., RTW-bench, Spadawan and BenchDW, for real-time, spatial and biological data warehouses, respectively. However, the benchmark's application span is necessarily quite narrow. One alternative is to resort to benchmark generators, also called tunable or generic benchmarks, such as OCB, DWEB or FlexBench, which help generate various data or workload models, and thus allow experiments to run in various conditions. The other alternative, which is preferred in recent benchmarks such as YSCB, OLTP-Bench and BigDataBench, is to offer a unifying framework that includes a comprehensive set of state-of-the-art benchmarks. However, the two latter approaches are mechanically detrimental to simplicity, which is a primordial criterion. It is thus necessary to devise benchmark suites that to not sacrifice simplicity too much.

FUTURE RESEARCH DIRECTIONS

The previous section showed that classical transaction and decision-oriented benchmarks are well established. However, big data benchmarking, which predominantly uses cloud technologies, faces a new paradigm and must measure new features. Thus, in addition to Gray's (1993) criteria for building a good benchmark, Folkerts et al. (2012) propose that the quality criteria that are commonly accepted by the benchmarking community must be revisited.

Although the cloud inherits from a long legacy of distributed systems, important issues are unique to the cloud. For instance, the concept of elasticity applied to data management may translate in the ability to bring in new data sources dynamically to meet emerging needs (Pedersen, 2010). Thus, cloud benchmark data models should be dynamic. Moreover, the three or four Vs (if we include

veracity) of big data are unequally addressed in current benchmarks, which mostly focus on scaling (volume) and, to a lesser extent, on variety with multi-benchmark suites such as BigDataBench. Among recent benchmarks, only HcBench features inter-job arrival rates that can simulate data streams. Yet, it is a quite low-level benchmark. Thus, RTDW-Bench and Bär and Golab's (2012) stream warehouse benchmark could be welcome additions in multi-benchmark suites.

With respect to veracity, specific security issues appeared in the new framework of the cloud, e.g., cloud provider or subcontractor espionage, cost-effective defense of availability or uncontrolled mashups (Chow et al., 2009). Such features are important to assess, for security is one of the top concern of cloud users and would-be users. Data consistency is also a concern, e.g., PRIMEBALL's metrics do not only target transaction performance and storage costs, but also data consistency. Bermbach et al. (2013) further advocate for a standard comprehensive benchmark for quantifying the consistency guarantees of eventually consistent storage systems. Moreover, for web application-based benchmarks, data quality assessment is also of premium importance. To the best of our knowledge, this intricate task has not been included yet in any big data benchmark.

Finally, the economic model of the cloud is fundamentally new. Instead of a costly initial investment, pay-as-you-go models allow users to pay a small amount per use, e.g., of a dataset, in return for a one-time advantage (Pedersen, 2010). Thus, cost is also a key criterion when benchmarking cloud/big data solutions. TPC benchmarks typically feature a cost metric, but it is presumably too high-level for fine-grained cost analyses.

CONCLUSION

Benchmarking is a small field, but it is nonetheless essential to data-centric research and industry. It serves both engineering and research purposes, when designing systems or validating solutions; as well as marketing purposes, when monitoring competition and comparing commercial products.

We subdivide benchmarks in three classes. First, standard, general-purpose benchmarks such as the TPC's do an excellent job in evaluating the global performance of systems. They are well suited to software selection by users and marketing battles by vendors, who try to demonstrate the superiority of their product at one moment in time. However, their relevance drops for some particular applications that exploit data or workloads models that are radically different from those they implement. Ad-hoc benchmarks are a solution. They either are adaptations of general-purpose benchmarks, or specifically designed benchmarks. Designing myriads of narrow-band benchmarks is not time-efficient, though, and trust in yet another new benchmark might prove limited in the community. Hence, the last alternative is to use generic or multi-benchmarks that feature a common framework for generating various experimental possibilities. The drawback of this approach is that benchmark complexity must be mastered. In conclusion, before starting a benchmarking experiment, users' needs must be carefully assessed so that the right benchmark or benchmark class is selected, and test results are meaningful.

It is nonetheless clear that the TPC plays a primordial role in the data benchmarking community, not only by issuing standards, but also by structuring and leading the community, e.g., by organizing the annual Technology Conference on Performance Evaluation and Benchmarking (TPCTC). This event does not only promote the TPC's activity, but also greatly encourages industrial and academic advances in the field of performance evaluation and benchmarking, whether they are related to the TPC or TPC benchmarks or not.

REFERENCES

<eref>Amplab. (2014). Big Data Benchmark. Retrieved from https://amplab.cs.berkeley.edu/benchmark/</eref>

<conf>Bär, A., & Golab, L. (2012). Towards benchmarking stream data warehouses. In *Proceedings of the 15th ACM International Workshop on Data Warehousing and OLAP (DOLAP 2012)* (pp. 105-112). Maui, USA: ACM.</conf>

<conf>Bermbach, D., Zhao, L., & Sakr, S. (2013). Towards Comprehensive Measurement of Consistency Guarantees for Cloud-Hosted Data Storage Services. In *Proceedings of the 5th Technology*

Conference on Performance Evaluation and Benchmarking (TPCTC 2013) (Vol. 8391, pp. 32-47). Riva del Garda, Italy: Springer, LNCS.</conf>

<conf>Carey, M. J., DeWitt, D. J., & Naughton, J. F. (1993). The OO7 benchmark. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD 1993)* (pp. 12-21). Washington, USA: ACM.</conf>

<conf>Carey, M. J., Dewitt, D. J., & Naughton, J. F. (1997). The BUCKY Object-Relational Benchmark. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD 1997)* (pp. 135-146). Tucson, USA: ACM.</conf>

<eref>Chen, Y., Alspaugh, S., Ganapathi, A., Griffith, R., & Katz, R. (2013). The Statistical Workload Injector for MapReduce (SWIM). Retrieved from https://github.com/SWIMProjectUCB/SWIM/wiki</eref>

<conf>Chow, R., Golle, P., Jakobsson, M., Shi, E., Staddon, J., Masuoka, R., & Molina, J. Controlling Data in the Cloud: Outsourcing Computation without Outsourcing Control. In *Proceedings of the1st ACM Cloud Computing Security Workshop (CCSW 2009)* (pp. 85-90). Chicago, USA: ACM.</conf>

<conf>Cooper, B. F., Silberstein, A., Tam, E., Ramakrishnan, R., & Sears, S. (2010). Benchmarking cloud serving systems with YCSB. In *Proceedings of the 1st ACM symposium on Cloud Computing (SoCC 2010)* (pp. 143-154). Indianapolis, USA: ACM.</conf>

<conf>Curino, C., Difallah, D. E., Pavlo, A., & Cudré-Mauroux, P. (2012). Benchmarking OLTP/Web Databases in the Cloud: the OLTP-Bench Framework. In *Proceedings of the 4th International Workshop on Cloud Data Management (CloudDB 2012)* (pp. 17-20). Maui, USA.</conf>

<jrn>Darmont, J., Bentayeb, F., & Boussaid, O. (2007). Benchmarking Data Warehouses. *International Journal of Business Intelligence and Data Mining*, *2*(1), 79-104.</jrn>

<jrn>Darmont, J., & Schneider, M. (2000). Benchmarking OODBs with a Generic Tool. *Journal of Database Management*, *11*(3), 16-27.</jrn>

<conf>Ferrarons, J., Adhana, M., Colmenares, C., Pietrowska, S., Bentayeb, F., & Darmont, J. (2013). PRIMEBALL: a Parallel Processing Framework Benchmark for Big Data Applications in the Cloud. In *Proceedings of the 5th Technology Conference on Performance Evaluation and Benchmarking (TPCTC 2013)* (Vol. 8391, pp. 109-124). Riva del Garda, Italy: Springer, LNCS.</conf>

<conf>Folkerts, E., Alexandrov, A., Sachs, K., Iosup, A., Markl, V., & Tosun, C. (2012). Benchmarking in the Cloud: What It Should, Can, and Cannot Be. *Selected Topics in Performance Evaluation and Benchmarking: 4th TPC Technology Conference, TPCTC 2012, Istanbul, Turkey, August 27, 2012, Revised Selected Papers* (Vol. 7755, pp. 173-188). Istanbul, Turkey: Springer, LNCS.</conf>

<edb>Gray, J. (Ed.). (1993). The Benchmark Handbook for Database and Transaction Processing Systems (2nd ed.). Morgan Kaufmann, San Francisco.</edb>

<conf>Huang, S., Huang, J., Dai, J., Xie, T., & Huang, B. (2010). The HiBench benchmark suite: Characterization of the MapReduce-based data analysis. In *Workshops Proceedings of the 26th International Conference on Data Engineering (ICDE 2010)* (pp. 41-51). Long Beach, USA.</conf>

<conf>Jedrzejczak, J., Koszlajda, T., & Wrembel, R. (2012). RTDW-bench: Benchmark for Testing Refreshing Performance of Real-Time Data Warehouse. In *Proceedings of the 23rd International Conference on Database and Expert Systems Applications (DEXA 2012)* (Vol. 7446, pp. 199-206). Vienna, Austria: Springer, LNCS.</conf>

<conf>Jia, Z., Zhan, J., Wang, L., & Zhang, L. (2013). DCBench: A benchmark suite for data center. Presented at the *19th IEEE International Symposium on High Performance Computer Architecture (HPCA 2013)* (Tutorial). Shenzhen, China.</conf>

<conf>Lee, S., Kim, S., & Kim, W. (2000). The BORD Benchmark for Object-Relational Databases. In *Proceedings of the 11th International Conference on Database and Expert Systems Applications (DEXA 2000)* (Vol. 1873, pp. 6-20). London, UK: Springer, LNCS.</conf>

<conf>Lopes Siqueira, T. L., Rodrigues Ciferri, R., Cesário Times, V., & Dutra de Aguiar Ciferri, C. Benchmarking Spatial Data Warehouses. In *Proceedings of the 12th International Conference on Data Warehousing and Knowledge Discovery (DAWAK 2010)* (Vol. 6263, pp. 40-51). Bilbao, Spain: Springer, LNCS.</conf>

<conf>Mahboubi, H., & Darmont, J. (2010). XWeB: The XML Warehouse Benchmark. In *Proceedings of the 2nd Technology Conference on Performance Evaluation and Benchmarking (TPCTC 2010)* (Vol. 6417, pp. 185-203). Singapore: Springer, LNCS.</conf>

Moussa, R. (2012). TPC-H Benchmark Analytics Scenarios and Performances on Hadoop Data Clouds. In Benlamri, R. (Ed.). *Networked Digital Technologies. Communications in Computer and Information Science* (Vol. 293, pp. 220-234). Springer.

<conf>O'Neil, P., O'Neil, E., Chen, X., & Revilak, S. (2009). The Star Schema Benchmark and Augmented Fact Table Indexing. In *Proceedings of the 1st Technology Conference on Performance Evaluation and Benchmarking (TPCTC 2009)* (Vol. 5895, pp. 237). Lyon, France: Springer, LNCS.</conf>

<eref>OLAP Council. (1998). APB-1 OLAP Benchmark Release II. Retrieved from http://www.olapcouncil.org/research/bmarkly.htm</eref>

<eref>Open Cloud Consortium. (2009). Generate synthetic site-entity log data for testing and benchmarking applications requiring large data sets. Retrieved from http://code.google.com/p/malgen/</eref>

<conf>Pavlo, A., Paulson, E., Rasin, A., Abadi, D.J., DeWitt, D.J., Madden, S., & Stonebraker, M. (2009). A comparison of approaches to large-scale data analysis. In *Proceedings of the of the ACM SIGMOD International Conference on Management of Data (SIGMOD 2009)* (pp. 165-178). Providence, USA.</conf>

<conf>Pedersen, T. B. (2010). Research challenges for cloud intelligence. In *Proceedings of the 2010 EDBT/ICDT Workshops.* Lausanne, Switzerland: ACM International Conference Proceeding Series.</conf>

<conf>Rabl, T., Frank, M., Danisch, M., Jacobsen, H.A., & Gowda, B. (2015). The Vision of BigBench 2.0. In *Proceedings of the 4th Workshop on Data analytics in the Cloud (DanaC 2015)* (pp. 3:1-3:4). Melbourne, Australia.</conf>

<jrn>Perriot, R., Pfeifer, J., D'Orazio, L., Bachelet, B., Bimonte, S., & Darmont, J. Cost Models for Selecting Materialized Views in Public Clouds. *International Journal of Data Warehousing and Mining, 10*(4), pp. 1-25.</jrn>

<edb>Saletore, V.A., Krishnan, K., Viswanathan, V., & Tolentino, M.E. (2013). HcBench: Methodology, Development, and Full-System Characterization of a Customer Usage Representative Big Data/Hadoop Benchmark. In *Proceedings of the 3rd and 4th Workshops on Big Data Benchmarking (WBDB 2013) – Revised selected papers* (Vol. 8585, pp. 73-93). Xi'an, China. San Jose, USA: Springer, LNCS.</edb>

<conf>Schmidt, K., Bächle, S., & Härder, T. (2009). Benchmarking Performance-Critical Components in a Native XML Database System. In *Proceedings of the 1st International Workshop on Benchmarking of XML and Semantic Web Applications (BenchmarX 2009)* (Vol. 5667, pp. 64-78). Brisbane, Australia: Springer, LNCS.</conf>

<eref>TPC. (2010). TPC Benchmark C Standard Specification Revision 5.11. Transaction Processing Performance Council. Retrieved from http://www.tpc.org</eref>

<eref>TPC. (2013). TPC Virtual Measurement Virtual System Standard Specification Version 1.2.0. Transaction Processing Performance Council. Retrieved from http://www.tpc.org</eref>

<eref>TPC. (2014a). TPC Benchmark H Standard Specification Revision 2.17.1. Transaction Processing Performance Council. Retrieved from http://www.tpc.org</eref>

<eref>TPC. (2014b). TPC Benchmark DI Standard Specification Version 1.1.0. Transaction Processing Performance Council. Retrieved from http://www.tpc.org</eref>

<eref>TPC. (2015a). TPC Benchmark E Standard Specification Version 1.14.0. Transaction Processing Performance Council. Retrieved from http://www.tpc.org</eref>

<eref>TPC. (2015b). TPC Benchmark DS Standard Specification Version 1.4.0. Transaction Processing Performance Council. Retrieved from http://www.tpc.org</eref>

<eref>TPC. (2015c). TPC Express Benchmark HS Standard Specification Version 1.3.0. Transaction Processing Performance Council. Retrieved from http://www.tpc.org</eref>

<conf>Triplet, T., & Butler, G. (2013). BenchDW: a generic framework for biological data warehouse benchmarking. In *Proceedings of the 28th ACM Symposium on Applied Computing (SAC 2013)* (pp. 1328-1334) Coimbra, Portugal: ACM.</conf>

<conf>Vranec, M., & Mlýnková, I. (2009). FlexBench: A Flexible XML Query Benchmark. In *Proceedings of the 14th International Conference on Database Systems for Advanced Applications (DASFAA 2009)* (Vol. 5463, pp. 421-435). Brisbane, Australia: Springer, LNCS.</conf>

<conf>Wang, L., Zhan, J., Luo, C., Zhu, Y., Yang, Q., He, Y., Gao, W., Jia, Z., Shi, Y., Zhang, S., Zheng, C., Lu, G., Zhan, K., Li, X., & Qiu, B. (2014). BigDataBench: A big data benchmark suite from internet services. In *Proceedings of the IEEE 20th International Symposium on High Performance Computer Architecture (HPCA 2014)* (pp. 488-499). Orlando, USA.</conf>

<conf>Yao, B. B., Özsu, T., & Khandelwal, N. (2004). XBench Benchmark and Performance Testing of XML DBMSs. In *Proceedings of the 20th International Conference on Data Engineering (ICDE 2004)* (pp. 621-633). Boston, USA.</conf>

<conf>Yasin, A., Ben-Asher, Y., & Mendelson, A. (2014). Deep-dive analysis of the data analytics workload in CloudSuite. In *Proceedings of the 2014 IEEE International Symposium on Workload Characterization (IISWC 2014)* (pp. 202-211). Raleigh, USA.</conf>

<conf>Zhang, X., Liu, K., Zou, L., Du, X., & Wang, S. (2011). Renda-RX: A Benchmark for Evaluating XML-Relational Database System. In *Proceedings of the 12th International Conference on Web-Age Information Management (WAIM 2011)* (Vol. 6897, pp. 578-589). Wuhan, China: Springer, LNCS.</conf>

ADDITIONAL READINGS

<conf>Alexandrov, A., Brücke, C., & Markl, V. (2013). Issues in big data testing and benchmarking. Presented at the *6th International Workshop on Testing Database Systems (DBTest 2013)*. New York, USA.</conf>

<jrn>Angles, R., Boncz, P., Larriba-Pey, J., Fundulaki, I., Neumann, T., Erling, O., Neubauer, P., Martinez-Bazan, N., Kotsev, V., & Toma, I. (2014). The linked data benchmark council: a graph and RDF industry benchmarking effort. *SIGMOD Record, 43*(1), 23-31.<jrn>

<conf>Armstrong, T.G., Ponnekanti, V., Borthakur, D., & Callaghan, M. (2013). LinkBench: a database benchmark based on the Facebook social graph. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD 2013)* (pp. 1185-1196). New York, USA.</conf>

<jrn>Boncz, P, Fundulaki, I., Gubichev, A., Larriba-Pey, J., & Neumann, T. (2013). The Linked Data Benchmark Council Project. *Datenbank-Spectrum, 13*(2), 121-129.<jrn>

<conf>Barata, M., Bernardino, J., & Furtado, P. (2014). Survey on Big Data and Decision Support Benchmarks. In *Proceedings of the 25th International Conference on Database and Expert Systems Applications (DEXA 2014)* (Vol. 8645, p. 174-182). Munich, Germany: Springer, LNCS.</conf>

<conf>Baru, C. K., Bhandarkar, M. A., Othayoth Nambiar, R., Poess, M., & Rabl, T. (2012). Setting the Direction for Big Data Benchmark Standards. In *Proceedings of the 4th Technology Conference on Performance Evaluation and Benchmarking (TPCTC 2012)* (Vol. 7755, pp. 197-208). Istanbul, Turkey: Springer, LNCS.</conf>

<jrn>Cheng, Y., & Rusu, F. Formal representation of the SS-DB benchmark and experimental evaluation in EXTASCID. *Distributed and Parallel Databases, 33*(3), pp. 277-317.</jrn>

<jrn>Difallah, D.E., Pavlo, A., Curino, C., & Cudré-Mauroux, P. (2013). OLTP-Bench: An Extensible Testbed for Benchmarking Relational Databases. *PVLDB, 7*(4), 277-288.</jrn>

<eref>Han, R., Jia, Z., Gao, W., Tian, X., & Wang, L. (2015). Benchmarking Big Data Systems: State-of-the-Art and Future Directions. Retrieved from http://arxiv.org/abs/1506.01494</eref>

<conf>Huppler, K. (2009). The Art of Building a Good Benchmark. In *Proceedings of the 1st Technology Conference on Performance Evaluation and Benchmarking (TPCTC 2009)* (Vol. 5895, pp. 18-30). Lyon, France: Springer, LNCS.</conf>

<conf>Iosup, A. (2013). IaaS cloud benchmarking: approaches, challenges, and experience. In *Proceedings of the 2013 international workshop on Hot topics in cloud services (HotTopiCS 2013)* (pp. 1-2). Prague, Czech Republic.</conf>

<eref>Jia, Z., Zhou, R., Zhu, C., Wang, L., Gao, W., Shi, Y., Zhan, J., & Zhang, L. (2013). The Implications of Diverse Applications and Scalable Data Sets in Benchmarking Big Data Systems. Retrieved from http://arxiv.org/abs/1307.7943</eref>

<eref>Kestelyn, J. (2014). Big Data Benchmarks: Toward Real-Life Use Cases. Retrieved from http://blog.cloudera.com/blog/2014/08/big-data-benchmarks-toward-real-life-use-cases/</eref>

<conf>Kuhlenkamp, J., Klems, M., & Röss, O. (2014). Benchmarking Scalability and Elasticity of Distributed Database Systems. In *Proceedings of the VLDB Endowment (VLDB 2014)* (Vol. 7, No. 12, pp. 1219-1230).</conf>

<jrn>Nambiar, R., Poess, M., Masland, A., Taheri, H. R., Emmerton, M., Carman, F., & Majdalany, M. (2013). TPC Benchmark Roadmap 2012. *Selected Topics in Performance Evaluation and Benchmarking, 7755*, pp. 1-20. doi:10.1007/978-3-642-36727-4_1</jrn>

<edb>Nambiar, R., Poess, M., Eds. (2015). *Performance Characterization and Benchmarking. Traditional to Big Data* (Vol. 8904): Springer, LNCS.</edb>

<conf>Qin, X., & Zhou, X. (2013). A Survey on Benchmarks for Big Data and Some More Considerations. In *Proceedings of the 14th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2013)* (Vol. 8206, pp. 619-627). Hefei, China: Springer, LNCS.</conf>

<edb>Rabl, T., Poess, M., Baru, C.K., & Jacobsen, H.A. (Eds.) (2012). *Proceedings of the 1st and 2nd Workshops on Big Data Benchmarking (WBDB 2012) – Revised selected papers* (Vol. 8163). San Jose, USA – Pune, India: Springer, LNCS.</edb>

<edb>Rabl, T., Jacobsen, H.A., Nambiar, R., Poess, M., Bhandarkar, M.A., & Baru, C.K. (Eds.) (2013). *Proceedings of the 3rd and 4th Workshops on Big Data Benchmarking (WBDB 2013) – Revised selected papers* (Vol. 8585). Xi'an, China – San Jose, USA: Springer, LNCS.</edb>

<edb>Rabl, T., Sachs, K., Poess, M., Baru, C.K., & Jacobsen, H.A. (Eds.) (2014). *Proceedings of the 5th Workshop on Big Data Benchmarking (WBDB 2014) – Revised selected papers* (Vol. 8991). Postdam, Germany: Springer, LNCS.</edb>

<conf>Rivero, C.R., Schultz, A., Bizer, C., & Ruiz, D. (2012). Benchmarking the Performance of Linked Data Translation Systems. Presented at the *Workshop on Linked Data on the Web (LDOW 2012)*. Lyon, France.</conf>

Surdu, S., Gripay, Y., Scuturici, V.M., Petit, J.M. (2013). P-Bench: Benchmarking in Data-Centric Pervasive Application Development. In *Transactions on Large-Scale Data- and Knowledge-Centered Systems XI* (vol. 8290, pp. 51-75). Springer, LNCS.

Waage, T., & Wiese, L. (2015). Benchmarking Encrypted Data Storage in HBase and Cassandra with YCSB. In *Foundations and Practice of Security* (Vol. 8930, pp. 311-325). Springer, LNCS.

Wang, H., Li, J., Zhang, Z., & Zhou, Y. (2014). Benchmarking Replication and Consistency Strategies in Cloud Serving Databases: HBase and Cassandra. In *Big Data Benchmarks, Performance Optimization, and Emerging Hardware* (Vol. 8807, pp. 71-82). Springer, LNCS.

KEY TERMS AND DEFINITIONS

Benchmark: A standard program that runs on different systems to provide an accurate measure of their performance.

Cloud Benchmarking: Use of cloud services in the respective (distributed) systems under test (Folkerts et al., 2012).

Database Benchmark: A benchmark specifically aimed at evaluating the performance of Database Management Systems (DBMSs) or DBMS components.

Data Model: In a data-centric benchmark, a database schema and a protocol for instantiating this schema, *i.e.*, generating synthetic data or reusing real-life data.

Performance Metrics: Simple or composite metrics aimed at expressing the performance of a system.

Synthetic Benchmark: A benchmark in which the workload model is artificially generated, as opposed to a real-life workload.

Workload Model: In a data-centric benchmark, a set of predefined read and write operations or operation templates to apply on the benchmark's database, following a predefined protocol.