Douglas Cunningham, Petra Hofstedt, Klaus Meer, Ingo Schmitt (Hrsg.): INFORMATIK 2015
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2015

Modelling the Performance, Energy Consumption and Efficiency of Data Management Systems

Raik Niemann[1] and Todor Ivanov[2]

Abstract: Many approaches, proposals and surveys to evaluate, measure and optimize the energy efficiency of data management systems were made during the last couple of years stressing the importance of this research field.

From the practitioner's perspective, for example developers and end users of data management systems, there is a big number of factors that influence the performance and energy efficiency of a particular data management system. For example, the replacement of hardware components or the surrounding operating system can have a significant impact. Both developers and end users put much effort into finding performance "bottlenecks", better hardware resource utilization and configurations. Besides, when it comes to a scale-out scenario, end users often face the situation to find a hardware configuration that offers both a reasonable performance and energy consumption, i.e. a resource planning.

Multiple evidence presented in this paper suggest that a model to simulate both the performance and energy consumption of a data management system is feasible. This allows to evaluate the energy efficiency by running a simulation rather than by performing real experiments. Compared to traditional ways, for example regression and compatibility tests, simulation runs are intended to drastically reduce the investments, both in time and hardware.

In this paper, both the reasons to create such simulation model as well as the characteristics and benefits of its fundamental design were discussed.

Keywords: Performance, Energy efficiency, Data management system

1 Introduction

The incentive to deal with the energy efficiency arises in many fields. When the number of incorporated components to handle a problem reaches a specific limit, the components themselves become a cost factor. Examples for such fields are data management and processing systems as well as sensor and mobile device networks.

To illustrate this issue, one can consider the development of data management systems in the last couple of years. The amount of information that is being created worldwide increases exponentially. The traditional data management and processing systems are not able to store and to process this amount efficiently. This leads to all the efforts that can

[1] Institute of Information Systems, Departement of Multimedial Information Systems, University of Applied Science Hof, Hof (Germany), `raik.niemann@iisys.de`
[2] Chair for Database and Information Systems, Institute for Informatics and Mathematics, Goethe University Frankfurt, Frankfurt am Main (Germany), `todor@dbis.cs.uni-frankfurt.de`

be summarized under the term “Big Data”. Examples are the appearance of *NoSQL* data management systems or data processing frameworks like *Hadoop*.

Although those systems allow the storage and processing of huge data amounts more efficiently, operators are still challenged. For example, there is a high number of combinations regarding the selection of operating systems, data management systems and hardware configurations. When it comes to a scale-out scenario or resource planning according to changed demands and requirements, tradeoffs are existing. The term “green computing” combines all efforts to find those tradeoffs.

It has been several years since the *Claramont report* [AAB+08] showed the directions, challenges and potential of “green computing”. The intention was to stimulate the research of new architectures, use cases and implementations of data management systems. Although some of them were realized, the reality shows that energy efficiency optimizations of data management systems are ambigous as outlined by *Wang et al.* [WFXS11].

The remaining paper is organized as follows: Section 2 briefly describes the calculation and methodology of the energy efficiency. Section 3 discusses issues with respect to the evaluation of the performance and energy consumption of data management systems. Section 4 addresses these issues and shows –as a solution– that a model for the simulation of both the performance and energy consumption is practical. Section 5 describes the fundamental design, the model and its key benefits whereas Section 6 presents the results of our first simulation models. Section 7 summarizes the paper.

2 Evaluation of the Energy Efficiency

There is a well known equation to calculate the energy efficiency *EE* of a data management system ([WFXS11], [XTW10]):

$$EE = \frac{W_{DB}}{W_e} = \frac{Perf \cdot t}{P \cdot t} = \frac{Perf}{P} \tag{1}$$

where W_{DB} is the useful work the data management system did and W_e the electrical work it consumed. P denotes the mean effective power. As W_{DB} is something arbitrary, it can not be expressed with a physical unit. In general W_{DB} is the equivalent of the mean performance $Perf$ over the time t.

As Equation 1 states, it is essential for the calculation of the energy efficiency to measure both the mean performance and the energy consumption. The former is given by a test methodology and the latter by a definition of the test apparatus. For comparability reasons, standards have been developed for both.

For instance, to evaluate the performance of a data management system, a benchmark implementing one or more real use cases, is run on top of it. There are widely accepted benchmarks for all kinds of data management systems. All these benchmarks are similar in that they report as a performance metric either the response time or the throughput. In general the response time is determined by executing a specific workload and measuring the time period from the initial access of the data management system until the end of the execution. This can be done simultaneously with a higher number of clients to evaluate the throughput.

For measuring the energy consumption while performing a benchmark, one or more energy consumption measurement devices are used. The test apparatus provides a description on how to measure the energy consumption. It also provides a list with the system components that should be measured.

3 Issues in the Evaluation of Energy Efficiency

There are several issues when it comes to the evaluation and optimization of the energy efficiency with respect to data management systems. Equation 1 states that any optimization follows either the increase of the performance or the reduction of the energy consumption.

For end users and developers of data management systems the optimization direction is important and still not clear [WFXS11]. Since the common way to evaluate and calculate the energy efficiency is by performing benchmarks while measuring the energy consumption, both groups have to invest into time (for the benchmark) and hardware components (for the benchmark running on different combinations of hardware components). As a result, all efforts lead to the *comparability* of data management systems in terms of performance, energy consumption and finally energy efficiency. The ability to compare two different data management systems or two different versions of the same one allows the decision making and the resource planning. For instance, developers of data management systems need comparability as the basis for regression tests. Similarly, end users need comparability to choose the more performant and energy efficient data management system for a given use case.

A general comparability of all known data management systems is impossible because this requires a general benchmark that covers all imaginable use cases and a common access interface to the data management systems in order to evaluate their performance. Additionally such a benchmark also requires a general methodology and test apparatus to measure the energy consumption for all possible combinations of hardware components.

Despite of that, it is feasible to compare data management systems that have a common access interface or a common domain. There are benchmarks for most domains that standardize the workloads to evaluate the performance of the tested data management systems. However, these benchmarks are not consistent regarding the reported metrics which makes the result comparision more difficult. A recent survey on benchmarks for *Big Data* [QZ13] extensively reviews the current metrics and discusses some of the characteristics and challenges they should address. An important point that the authors make is the need of a complete, end-to-end benchmarking suite, including both component-based and application-oriented benchmarks along with critical metrics such as energy consumption. On the contrary, more general metrics which are independent of workload type or based on processor micro-architecture characteristics are reported. Examples are Cycles per Instructions (CPI), first level data cache misses per 1000 instructions (L1 MPKI) and last level cache (LLC) miss ratio, presented in [XYB+13] and [DKL+13]. Finally, important new metrics like data processed per second and data per Joule are proposed by [LZJ+12] to better measure data processing and energy consumption.

In contrast to the evaluation of the performance, the methodology to measure the consumed electrical energy according to Equation 1 is not consistent. In particular we refer to the following issues:

- There are only two major standards that are widely accepted: *SPECpower* [(SP15] and *TPC-Energy* [(TP15]. In contrast to *TPC-Energy* which standardizes an additional metric for their *TPC-*(*C,E,H,DS*) benchmark, *SPECpower* is a more general way to relate the performance to the energy consumption.
- The set of components to be measured is not consistent. *TPC-Energy* as well as *SPECpower* define the components of a system under test (SUT) but some of them are only optional. However, the reality shows that only the measurement of the overall power consumption matters. This is based on the fact that both *TPC-Energy* and *SPECpower* require the usage of a separate software that is responsible for the entire benchmark execution and also to measure the energy consumption of the incorporated devices. In fact, this software supports only a couple of measurement devices.
- Another important aspect is the inconsistent definition of the time frame that should be measured. *TPC-Energy* defines this as the time period that it takes to execute the benchmark. *SPECpower* extends this time period with short time periods before and after the benchmark (so called "ramp-up"). The reason for this is to reflect the preparation of the benchmark.

4 Reasons for a Model

In our opinion all approaches that were developed in the last decade towards the optimization of the energy efficiency in data management systems showed the same empirical progress and semantics as the ones known from the behavioral science.

This means that three stages are required to fully understand a topic. In the first stage all factors that could have a potential influence are gathered. In the second stage the influence factors are tested individually to estimate their impact. In the final third stage a model abstracts all findings and allows the deduction of new knowledge.

With respect to the topic of this paper it is our belief that enough research results were published to finally form a model. To prove our point we surveyed prominent publications and approaches.

The first research concerning the factors that could have a potential influence on the energy efficiency were evaluated and presented by *Xu et al.* [XTW10], *Wang et al.* [WFXS11] and *Tsirogiannis et al.* [THS10]. Although they focused on relational data management systems, important influence factors were identified and first measurement methodologies were proposed. In addition to this the assumption was verified that the optimization of single influence factors have a significant impact on the energy efficiency. *NoSQL* data management systems became very popular, especially in conjunction with the *Hadoop* ecosystem. This introduced additional potential influence factors as presented by *Lang et al.* [LKP11], *Abadi* [Aba12], *Rabl et al.* [RGVS+12] and *Baru et al.* [BBN+13].

The impact of the identified factors was evaluated by executing benchmarks for fixed combinations of those factors, as reported for example by *Vasic et al.* [VBSK09], *Lang and Patel* [LP09] and *Poess et al.* [PNV+10]. Prominent approaches modified existing data management systems or their test apparatus to optimize either the performance (response time and throughput) or the energy consumption of the system. *Hindman et al.* [HKZ+11] tried to get existing data management systems aware of the energy consumption which is in return just another aspect of optimizing for the performance. This was done by combining existing heuristics like indices and query planner statistics with energy consumption metrics. Another approach was to take energy efficiency into consideration when the data management software is in its planning and implementation stage (so called *software performance engineering* as described in [WFP07] and [DM02]). A prominent example for software performance engineering is *F1* [SOE+12], a relational data management system developed by *Google* addressing the energy efficiency and performance as key requirements.

In addition to the given publications from the academia, even more white papers were released to the public by the developers of modern data management systems. Those white papers present experimental performance results as well as architectural insights (for example [Dat14a] and [Dat14b] for *Cassandra*).

5 Model Characteristics

As implied by Equation 1, a model has to simultaneously consider both the performance and energy consumption aspects of a system. To achieve this, both aspects need to be abstracted and generalized. In particular this means to find similarities that data management systems, benchmarks and benchmark methodologies have in common.

First, we studied the similarities of prominent data management systems. The result was that they only share the basic fundamental architecture ([WFXS11], [WV01]). In principle they offer an interface to save and relate facts or information of a specific domain with respect to the *ACID* properties (atomicity, consistency, isolation and durability). They also offer an interface to query the data in an efficient and performant manner by using heuristics, for example by using different indexing mechanisms.

Second, we studied major benchmarks in order to find their similarities. To get an overview how the major benchmarks define their performance metric, we surveyed them according to their reported metric. A summary of metrics is presented in Table 1. It is noticeable that the dominant metric is either throughput or response time. The latter can be expressed as throughput if the queried data amount is known. Next, we reconsidered all approaches given in the previous sections as well as the similarities mentioned above to find a model that fully supports all aspects. We came to the conclusion that a basic data flow model is the most promising solution.

Benchmark	Category and metric
TPC-E	OLTP processing workload of a brokerage firm Transactions per second
TPC-C	OLTP processing workload for heavy transactions Transactions per minute
TPC-H	OLAP decision support workload for large sets of data Query response time
TPC-DS	Like TPC-H but with more I/O and CPU load Query response time
TPC-VMS	Additional methodology for virtualized TPC-(E,C,H,DS) Summarized throughput
BigData benchmark	OLAP benchmark for MPP architectures [AMP14] Query response time
BigDataBench	Compound OLTP and OLAP benchmark for various architectures [WZL+14] Throughput
HiBench	OLAP benchmark for *Hadoop*/*Hive* [HHD+10] Throughput and response time
BigBench	TPC-DS inspired benchmark [GRH+13] Throughput (queries per hour)
LinkBench	Social graph database benchmark [APBC13] Query response times
YCSB	Framework to benchmark cloud services, e.g. DBMS' [CST+10] Operations per second
XMark	Benchmark for XML operations in a data management system [SWK+02] Query response time
Cloudsuite	Compound benchmark for datacenter operations [FAK+12] (undecided)
TPCx-HS	Benchmark to stress *Hadoop* configurations Response time and throughput

Tab. 1: Surveyed data management system benchmarks with their metrics

The main reason for a general model is the fact that the data flow is fundamental and valid for all types of software that process data in a structured manner such as data management systems which share the same architecture. Furthermore, a model that is specifically designed to reflect the common data flow within a data management system has also the advantage that performance can be simulated and predicted. *Bontempi and Kruijtzer* [BK02], *Kounev et al.* [KGS08] and *Nambiar and Poess* [NP10] showed that it is possible to simulate and precisely predict the performance of a system.

Additionally the amount or structure of the input data for the data flow model does not matter. This is most useful with regard to benchmarks. Since all the surveyed benchmarks generate a predefined amount of domain specific data, this data can be used as input for the model. The advantage of this approach is that the simulation results can be directly compared to experimental results. The surveyed benchmarks have also in common that they focus on the evaluation of either the response time or the throughput. With respect to the data flow model both metrics can be measured and reported.

Another reason for forming a model based on the data flow is the possibility to use additional metrics. According to *Dietrich et al.* [DBCRW92], a data flow model that includes all components required for processing data could also consider all hardware components. When it comes to the simulation of the data flow, an additional metric like the mean power consumption per simulated hardware component allows to calculate the overall power consumption.

In a nutshell, a data flow model whose components reflect the common architecture of data management systems as well as all the incorporated processing hardware is able to simulate the data flow that was previously created by a benchmark. In conjunction with an additional metric like the mean power consumption per component, the simulation of the overall power consumption is possible. Both the performance simulation results and the overall power consumption allow the calculation of the energy efficiency defined in Equation 1.

The key benefit of this model is the significant reduction in time and hardware investments. As outlined in Section 2 the common way to evaluate both the performance and energy consumption aspects is to execute a benchmark for comparability purposes. This needs to be done on different hardware configurations for a holistic evaluation of the energy efficiency. Simulating those aspects on the basis of the model would be faster without the need to invest into the desired hardware configuration.

From the practitioner's perspective, both developers and end users of data management systems would benefit from the proposed model. A variety of use cases are possible. They have all in common that only three basic simulation parameters are required: the data management system, the hardware configuration and the amount or domain of the data to be processed. Fixing two of this three parameters while varying the remaining one allows to simulate the effects on the performance and energy consumption.

For example, it enables developers to simulate the effects of possible implementation changes in the architecture of a data management system such as the replacement of the data storage layout or indices. This replacement can be made in the model instead of the software implementation. A simulation with a given amount of data and hardware configuration demonstrates the effects on the performance without having the architectural change actually implemented and evaluated by performing a benchmark. Moreover, the usual process of doing regression tests can be omitted.

In contrast to developers, end users of existing data management systems like datacenter operators are more interested in resource planning or scale-out scenarios. Given an existing data management system and hardware configuration, the effects on the performance and energy consumption can be simulated while changing the amount of data to be processed. The results can be used for further simulations, for example to find a hardware configuration that is more performant and energy-saving at the same time. Similarly, the most energy efficient data management system can be identified by simulation if the hardware configuration and the data size are known in advance.

6 Model Implementation and First Results

In [NI15], the data flow of a data management system was implemented by using *Queued Petri Nets* (QPN[1]). This model is intended to reflect the data flow of a data management system among all components that could have an impact on the performance. This includes the surrounding operating system, technical components (main memory, mass storage and CPU) and the common components of a data management system (access control, query plan generation and optimization, query plan isolation and query plan execution).

The basic model was evaluated and proven by simulating two prominent database benchmarks (*TPC-H* and *StarSchema Benchmark*). The simulation runs reported the simulated query response time and statistical values for all components of the model, for example the throughput of transitioned marks per simulated second. The throughput is important when it comes to the calculation of the simulated energy consumption. The simulated query execution times, energy consumption and efficiency values were compared with the experimental ones that were presented in [NKZG13]. On average the experimental and simulated values for the *TPC-H* benchmark differ by 36.55 percent for the query execution times and by 21.91 percent for the energy efficiency, respectively. For the *StarSchema Benchmark*, the experimental and simulated values differ by 19.77 percent for the query execution times and by 20.77 percent for the energy efficiency, respectively.

However, the usage of the basic model is limited because simulation runs aim to reflect the data flow of a data management system running on a single server machine. As a result, it only allows to simulate and to study the effects of vertical scale-out scenarios.

To simulate the data flow of a data management system that is distributed over several machines, it was necessary to introduce a new QPN model that enables the existing models to exchange data in the form of marks. In other words, network functionality was added to the basic models in [Nie15]. Simulation runs using the enhanced models in combination with the new introduced model allow additionally to observe the effects of horizontal scale-out scenarios. The enhanced models are able to simulate any computer networks. This means that the topology, the data transmission bandwidth and performance can be taken into consideration.

To prove the enhanced models, experiments on a bladecenter with seven uniform blades were performed. The experiments used the *Yahoo Cloud Serving Benchmark* (YCSB, [CST$^+$10]) on the the distributed data management system *Cassandra*. The enhanced models were set up and adjusted to reflect both the used real test apparatus and the benchmark. A comparison between the real and simulated experimental results showed a difference of 20.07 percent on average for the used YCSB workloads and 47.25 percent on average for

[1] A QPN combines Colored Petri Nets (CPNs), Generalized Stochastic Petri Nets (GSPNs) and queuing principles as well as scheduling strategies to the places. The formal definition for a QPN is fairly extensive and can be found in [KB03].

the energy efficiency, respectively. Note that to the best of our knowledge there is no other QPN model that is able to simulate the performance and energy consumption for the tested benchmarks. Therefore a comparison of the accuracy is infeasible.

In contrast to the experiments using the basic model mentioned before, the YCSB benchmark in combination with *Cassandra* turned out to be less complex including fewer transition rule and mark sets. This enabled the simulation of the workloads to finish in 3 minutes and 46 seconds on average to get a simulated response time of one YCSB workload. Compared to the experimental workload response times, this is nearly 8000 times faster.

The simulation model allowed to perform additional runs for *Cassandra* clusters having more than seven nodes. As a result, the simulation results for a given use case (fixed data size of 300 GByte to be stored in the *Cassandra* cluster) revealed that the optimal number of nodes is 10. This means that the performance and energy consumption gains fell under 10 percent with every subsequently added *Cassandra* node.

Further simulation runs showed a turning point for a *Cassandra* cluster having 25 nodes. At this point the simulated communication between the cluster nodes had an impact on the performance. As a result, the performance decreased by nearly 2.1 percent with every subsequently added *Cassandra* node. Note that these additional simulation results can not be validated with real experiments since the used test apparatus had only seven nodes.

7 Conclusion

In our opinion there is enough academic research and experimental results to form a model that is capable of simulating and predicting both the performance and the energy consumption of a data management system. This allows to calculate the energy efficiency in order to optimize it.

Based on our survey, multiple reasons motivating why such a model is feasible were outlined. Next we introduced the fundamental components and characteristics of the model, together with its key benefits and advantages. In [NI15] and [Nie15], we presented an approach to simulate the data flow of three prominent benchmarks (*TPC-H*, *StarSchema* and *Yahoo Cloud Serving Benchmark*) for two popular data management systems (*PostgreSQL* and *Cassandra*). The data flow was modelled using *Queued Petri Nets*. The executed simulation runs allowed to predict the response time of the benchmark workloads and to calculate the overall energy consumption for the simulated hardware components.

The simulation results were compared with experimental ones and showed a difference of 25.46 percent on average for the query or workload response times and 29.97 percent on average for the energy efficiency, respectively. The simulation runs were up to 8.000 times faster when compared to the real benchmark executions on the test servers. In [Nie15], additional simulation runs with changed hardware configuration revealed the turning point between the performance and energy consumption for a *Cassandra* cluster. In this way the most energy efficient hardware configuration for a given use case was found.

References

[AAB+08] Rakesh Agrawal, Anastasia Ailamaki, Philip A Bernstein, Eric A Brewer, Michael J Carey, Surajit Chaudhuri, AnHai Doan, Daniela Florescu, Michael J Franklin, Hector Garcia-Molina, et al. The Claremont report on database research. *ACM SIGMOD Record*, 37(3):9–19, 2008.

[Aba12] Daniel Abadi. Consistency tradeoffs in modern distributed database system design: CAP is only part of the story. *Computer*, 45(2):37–42, 2012.

[AMP14] AMPLab. BigData benchmark, 2014. https://amplab.cs.berkeley.edu/benchmark/.

[APBC13] Timothy G Armstrong, Vamsi Ponnekanti, Dhruba Borthakur, and Mark Callaghan. Linkbench: a database benchmark based on the facebook social graph. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pages 1185–1196. ACM, 2013.

[BBN+13] Chaitanya Baru, Milind Bhandarkar, Raghunath Nambiar, Meikel Poess, and Tilmann Rabl. Benchmarking Big Data Systems and the BigData Top100 List. *Big Data*, 1(1):60–64, 2013.

[BK02] Gianluca Bontempi and Wido Kruijtzer. A data analysis method for software performance prediction. In *Proceedings of the conference on Design, automation and test in Europe*, page 971. IEEE Computer Society, 2002.

[CST+10] Brian F Cooper, Adam Silberstein, Erwin Tam, Raghu Ramakrishnan, and Russell Sears. Benchmarking cloud serving systems with YCSB. In *Proceedings of the 1st ACM symposium on Cloud computing*, pages 143–154. ACM, 2010.

[Dat14a] DataStax. Apache Cassandra 2.0 Documentation, 2014. http://docs.datastax.com/en/cassandra/2.0/pdf/cassandra20.pdf.

[Dat14b] DataStax. DataStax Enterprise Reference Architecture, 2014. http://www.datastax.com/wpcontent/uploads/2014/01/WPDataStaxEnterpriseReference-Architecture.pdf.

[DBCRW92] Suzanne W. Dietrich, M Brown, Enrique Cortes-Rello, and S Wunderlin. A practitioner's introduction to database performance benchmarks and measurements. *The Computer Journal*, 35(4):322–331, 1992.

[DKL+13] Martin Dimitrov, Karthik Kumar, Patrick Lu, Vish Viswanathan, and Thomas Willhalm. Memory system characterization of big data workloads. In *Big Data, 2013 IEEE International Conference on*, pages 15–22. IEEE, 2013.

[DM02] Elizabeth D Dolan and Jorge J More. Benchmarking optimization software with performance profiles. *Mathematical programming*, 91(2):201–213, 2002.

[FAK+12] Michael Ferdman, Almutaz Adileh, Onur Kocberber, Stavros Volos, Mohammad Alisafaee, Djordje Jevdjic, Cansu Kaynak, Adrian Daniel Popescu, Anastasia Ailamaki, and Babak Falsafi. Clearing the clouds: a study of emerging scale-out workloads on modern hardware. In *Proceedings of the seventeenth international conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '12, pages 37–48, New York, NY, USA, 2012. ACM.

[GRH+13] Ahmad Ghazal, Tilmann Rabl, Minqing Hu, Francois Raab, Meikel Poess, Alain Crolotte, and Hans-Arno Jacobsen. BigBench: towards an industry standard benchmark for big data analytics. In *Proceedings of the 2013 ACM SIGMOD international conference on Management of data*, pages 1197–1208. ACM, 2013.

[HHD+10] Shengsheng Huang, Jie Huang, Jinquan Dai, Tao Xie, and Bo Huang. The HiBench benchmark suite: Characterization of the MapReduce-based data analysis. In *Data Engineering Workshops (ICDEW), 2010 IEEE 26th International Conference on*, pages 41–51. IEEE, 2010.

[HKZ+11] Benjamin Hindman, Andy Konwinski, Matei Zaharia, Ali Ghodsi, Anthony D Joseph, Randy H Katz, Scott Shenker, and Ion Stoica. Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center. In *NSDI*, volume 11, pages 22–22, 2011.

[KB03] Samuel Kounev and Alejandro Buchmann. Performance modelling of distributed e-business applications using Queuing Petri Nets. In *Performance Analysis of Systems and Software, 2003. ISPASS. 2003 IEEE International Symposium on*, pages 143–155. IEEE, 2003.

[KGS08] Samuel Kounev, Ian Gorton, and Kai Sachs, editors. *Performance Evaluation: Metrics, Models and Benchmarks*, volume 5119, Darmstadt, Germany, June 2008. SPEC International Performance Evaluation Workshop, Springer.

[LKP11] Willis Lang, Ramakrishnan Kandhan, and Jignesh M Patel. Rethinking Query Processing for Energy Efficiency: Slowing Down to Win the Race. *IEEE Data Eng. Bull.*, 34(1):12–23, 2011.

[LP09] Willis Lang and Jignesh Patel. Towards eco-friendly database management systems. *arXiv preprint arXiv:0909.1767*, 2009.

[LZJ+12] Chunjie Luo, Jianfeng Zhan, Zhen Jia, Lei Wang, Gang Lu, Lixin Zhang, Cheng-Zhong Xu, and Ninghui Sun. CloudRank-D: benchmarking and ranking cloud computing systems for data processing applications. *Frontiers of Computer Science*, 6(4):347–362, 2012.

[NI15] Raik Niemann and Todor Ivanov. Evaluating the Energy Efficiency of Data Management Systems. In *4th International Workshop on Green and Sustainable Software, ICSE 2015*. IEEE, 2015.

[Nie15] Raik Niemann. Evaluating the Performance and Energy Consumption of Distributed Data Management Systems. In *1st Green in Software Engineering Workshop, ICGSE 2015*. IEEE, 2015.

[NKZG13] Raik Niemann, Nikolaos Korfiatis, Roberto Zicari, and Richard Göbel. Evaluating the Energy Efficiency of OLTP Operations. In *Availability, Reliability, and Security in Information Systems and HCI*, pages 28–43. Springer, 2013.

[NP10] Raghunath Nambiar and Meikel Poess, editors. *Performance Evaluation, Measurement and Characterization of Complex Systems*, volume 6417, Singapore, September 2010. Second TPC Technology Conference, TPCTC, Springer.

[PNV+10] Meikel Poess, Raghunath Othayoth Nambiar, Kushagra Vaid, John M Stephens Jr, Karl Huppler, and Evan Haines. Energy benchmarks: a detailed analysis. In *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking*, pages 131–140. ACM, 2010.

[QZ13] Xiongpai Qin and Xiaoyun Zhou. A Survey on Benchmarks for Big Data and Some More Considerations. In *Intelligent Data Engineering and Automated Learning–IDEAL 2013*, pages 619–627. Springer, 2013.

[RGVS+12] Tilmann Rabl, Sergio Gómez-Villamor, Mohammad Sadoghi, Victor Muntés-Mulero, Hans-Arno Jacobsen, and Serge Mankovskii. Solving big data challenges for enterprise application performance management. *Proceedings of the VLDB Endowment*, 5(12):1724–1735, 2012.

[SOE+12] Jeff Shute, Mircea Oancea, Stephan Ellner, Ben Handy, Eric Rollins, Bart Samwel, Radek Vingralek, Chad Whipkey, Xin Chen, Beat Jegerlehner, Kyle Littlefield, and Phoenix Tong. F1 - The Fault-Tolerant Distributed RDBMS Supporting Google's Ad Business. In *SIGMOD*, 2012. Talk given at SIGMOD 2012.

[(SP15] Standard Performance Evaluation Corporation (SPEC). SPEC Power and Performance Methodology, 2015. https://www.spec.org/power/docs/SPEC-Power_and-_Performance _Methodology.pdf.

[SWK+02] Albrecht Schmidt, Florian Waas, Martin Kersten, Michael J Carey, Ioana Manolescu, and Ralph Busse. XMark: A benchmark for XML data management. In *Proceedings of the 28th international conference on Very Large Data Bases*, pages 974–985. VLDB Endowment, 2002.

[THS10] Dimitris Tsirogiannis, Stavros Harizopoulos, and Mehul A Shah. Analyzing the energy efficiency of a database server. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 231–242. ACM, 2010.

[(TP15] Transaction Processing Performance Council (TPC). TPC Energy Specification, 2015. http://www.tpc.org/TPC_Documents_Current_Versions/pdf/TPC Energy Specification 1.5.0.pdf.

[VBSK09] Nedeljko Vasić, Martin Barisits, Vincent Salzgeber, and Dejan Kostic. Making cluster applications energy-aware. In *Proceedings of the 1st workshop on Automated control for datacenters and clouds*, pages 37–42. ACM, 2009.

[WFP07] Murray Woodside, Greg Franks, and Dorina C Petriu. The future of software performance engineering. In *Future of Software Engineering, 2007. FOSE'07*, pages 171–187. IEEE, 2007.

[WFXS11] Jun Wang, Ling Feng, Wenwei Xue, and Zhanjiang Song. A survey on energy-efficient data management. *ACM SIGMOD Record*, 40(2):17–23, 2011.

[WV01] Gerhard Weikum and Gottfried Vossen. *Transactional information systems: theory, algorithms, and the practice of concurrency control and recovery*. Elsevier, 2001.

[WZL+14] Lei Wang, Jianfeng Zhan, Chunjie Luo, Yuqing Zhu, Qiang Yang, Yongqiang He, Wanling Gao, Zhen Jia, Yingjie Shi, Shujie Zhang, et al. Bigdatabench: a big data benchmark suite from internet services. In *High Performance Computer Architecture (HPCA), 2014 IEEE 20th International Symposium on*, pages 488–499. IEEE, 2014.

[XTW10] Zichen Xu, Yi-Cheng Tu, and Xiaorui Wang. Exploring power-performance tradeoffs in database systems. In *Data Engineering (ICDE), 2010 IEEE 26th International Conference on*, pages 485–496. IEEE, 2010.

[XYB+13] Wen Xiong, Zhibin Yu, Zhendong Bei, Juanjuan Zhao, Fan Zhang, Yubin Zou, Xue Bai, Ye Li, and Chengzhong Xu. A characterization of big data benchmarks. In *Big Data, 2013 IEEE International Conference on*, pages 118–125. IEEE, 2013.