Application-Level Benchmarking of Big Data Systems

Chaitan Baru

San Diego Supercomputer Center,
University of California San Diego
baru@sdsc.edu

Tilmann Rabl

bankmark
Germany
tilmann.rabl@bankmark.de

Abstract. The increasing possibilities to collect vast amounts of data—whether in science, commerce, social networking, or government—have led to the "big data" phenomenon. The amount, rate, and variety of data that are assembled—for almost any application domain—is necessitating a re-examination of old technologies and development of new technologies to get value from the data, in a timely fashion. With increasing adoption and penetration of mobile technologies, and increasing ubiquitous use of sensors and small devices in the so-called *Internet of Things*, the big data phenomenon will only create more pressures on data collection and processing for transforming data into knowledge for discovery and action.

A vibrant industry has been created around the big data phenomena, leading also to an energetic research agenda in this area. With the proliferation of big data hardware and software solutions in industry and research, there is a pressing need for benchmarks that can provide objective evaluations of alternative technologies and solution approaches to a given big data problem. This chapter gives an introduction to big data benchmarking and presents different proposals and standardization efforts.

1 Introduction

As described in [8], database system benchmarking has a rich history, from the work described in the paper entitled, *A Measure of Transaction Processing Power* [10], to the establishment of the Transaction Processing Council in 1988, and continuing into the present. The pressing need for database benchmark standards was recognized in the mid to late 1980's, when database technology was relatively new, and a number of companies were competing directly in the database systems software marketplace. The initial efforts were simply on persuading competing organizations to utilize the *same* benchmark specification—such as the *DebitCredit* benchmark introduced in [11]. However, the benchmark results were published directly by each company, often eliminating key requirements specified in the original benchmark. Thus, there was need for a standards-based approach, along with a standards organization that could uniformly enforce benchmark rules while also certifying results produced, thereby providing a stamp of approval on the results. Thus was born the *Transaction Processing Performance Council*, or *TPC*, in 1988. With its early benchmarks, such as TPC-C, TPC-D, and TPC-H, the TPC was successful in producing widely used benchmark standards that have led directly to database system product improvements and made a real impact on product features.

adfa, p. 1, 2011.

With the rapid growth in big data[1] applications, and vendor claims of hardware and software solutions aimed at this market, there is once again a need for objective benchmarks for systems that support big data applications. In the emerging world of "big data", organizations were once again publishing private benchmark results, which claimed performance results that were not audited or verified. To address this need, a group from academia and industry (including the authors), organized the first *Workshop on Big Data Benchmarking (WBDB)* in May 2012, in San Jose, California. Discussions at the WBDB workshop covered the full range of issues, and reinforced the need for benchmark standards for big data. However, there was also recognition of the challenge in defining a commonly agreed upon set of big data application scenarios that could lead towards benchmark standards.

In retrospect, the early TPC benchmarks had an easier time in this regard. Their initial focus was transaction processing—typically defined by *insert, update, delete*, and *read* operations on records or fields within records. Examples are point-of-sale terminals in retail shopping applications, bank teller systems, or ticket reservation systems. Subsequent benchmarks extended to SQL query processing with relational database systems. Big data applications scenarios are, however, much more varied than transaction processing plus query processing. They may involve complex transformation of data, graph traversals, data mining, machine learning, sequence analysis, time series processing, and spatiotemporal analysis, *in addition to* query processing. The first challenge, therefore, is to identify application scenarios that capture the key aspects of big data applications. Application-level data benchmarks are *end-to-end benchmarks* that strive to cover the performance of all aspects of the application, from data ingestion to analysis.

A benchmark specification must pay attention to multiple aspects, including:

(a) The so-called *system under test (SUT),* i.e., the system or components that are the focus of the testing, which may range from a single hardware or software component to a complete hardware or software systems.

(b) The types of *workloads*, from application-level to specific component-level operations. *Component benchmarks* focus on specific operations, examples are I/O system benchmarks, graphics hardware benchmarks, or sorting benchmarks. The types of workloads range from very simple *micro-benchmarks*, which test a certain type of operation, to complex application simulations or replay of real workloads.

(c) The *benchmarking process*. *Kit-based benchmarks* provide an implementation or suite of tools that automates the benchmarking process. *Specification-based benchmarks* describe the detailed benchmarking process and allow for different implementations of the benchmark. The former are typically use in component benchmarks, while the latter are used for database, end-to-end benchmarks.

(d) The *target audience*. Benchmark details and especially the representation of the results may differ depending upon the target audience for the benchmark. End users and product marketing may require results that are easily comparable with realistic

[1] The term "big data" is often written with capitals, i.e. Big Data. In this paper, we have chosen to write this term without capitalization.

workloads. Performance engineers may prefer workloads that cover typical modes of operation. System testers may want to cover *all* modes of operation, but also need deep insights into the system behavior. Big data benchmarks exist in all of the forms described above.

While benchmarks can only *represent* real-world scenarios—and are not the real world scenarios themselves—they nonetheless play an essential role. They can represent a broad class of application needs, requirements, and characteristics; provide repeatability of results; facilitate comparability among different systems; and provide efficient implementations. A good benchmark would represent the important aspects of real world application scenarios as closely as possible, provide repeatability and comparability of results, and would be easy to execute.

The rest of this paper is structured as follows. Section 2 provides examples of some big data application scenarios. Section 3 describes useful benchmark abstractions that represent large classes of big data applications. Section 4 describes the approach taken by different benchmark standards, such as TPC and SPEC. Section 5 describes current benchmarking efforts, and Section 6 provides a conclusion.

2 Big Data Application Examples

While there is broad consensus on the potential for big data to provide new insights in scientific and business applications, characterizing the particular nature of big data and big data applications is a challenging task—due to the breadth of possible applications. In 2014, the US *National Institute for Standards and Technologies (NIST)* initiated a NIST Big Data Public Working Group (NBD-PWG) in order to tackle the issue of developing a common framework and terminology for big data and big data applications[2]. As documented in the NBD-PWG volume on *Use Cases and General Requirements*[3], the range of real-world big data applications is broad, and includes collection and archiving of data; use in trading and financial sector; delivery of streaming content in a variety of applications scenarios including, for example, security, entertainment, and scientific applications; indexing to support web search; tracking data streams related to shipping and delivery of physical items (e.g., by FedEX, UPS, or US Postal Service); collection and analysis of data from sensors, in general; personalized health, Precision Medicine and other applications in healthcare and Life Sciences (including electronic medical records, pathology, bio-imaging, genomics, epidemiology, people activity models, and biodiversity); deep learning with social media and a variety of other data; processing for driverless cars; language translation; smart grids; and others.

The NIST Use Cases, mentioned earlier, provide an example application related to the *Genome in a Bottle Consortium*, which requires integration of data from multiple sequencing technologies and methods; development of robust characterization of whole

[2] NIST Big Data Public Working Group, http://bigdatawg.nist.gov/home.php

[3] NIST NBD-PWG Use Cases and Requirement, http://bigdatawg.nist.gov/usecases.php

human genomes as reference materials; and, development of methods to use these reference materials to assess performance of any genome sequencing run. A current pilot application at NIST employing open-source bioinformatics sequencing software on a 72-core cluster has generated 40TB of data. However, DNA sequencers will be able to generate ~300GB compressed data per day in the near future. An individual lab will easily be able to produce petabytes of genomics data in future.

In another example from the US Census Bureau, it is noted that, since the costs of conducting demographic surveys are increasing even as survey responses decline, the Census Bureau and other such survey-oriented organizations will, in future, consider the use of advanced techniques for demographic surveys, including recommendation systems to improve response rates; the use of "mash ups" of data from multiple sources; and the use of historical survey "para-data", i.e., administrative data about the survey itself, to help improve operational processes and data quality. The end goal is to increase the overall quality and reduce the cost of field surveys. In the current approach, the US Census gathers about 1 petabyte of data from surveys and other government administrative sources. During the decennial census period, these data are streamed into the system, with approximately 150 million records transmitted as field data. In future, analytic techniques will need to be developed to provide statistical estimations that provide more detail on a more near real-time basis for less cost. Data quality needs to be high and must be statistically checked for accuracy and reliability throughout the collection process. The reliability of estimated statistics from "mashed up" sources will need to be evaluated. All processes must be auditable for security and confidentiality as required by various legal statutes and, throughout the process, all data must remain confidential and secure.

A third use case deals with *large-scale deep learning* models. Neural networks with many more neurons and connections combined with large datasets are increasingly the top performers in benchmark tasks for vision, speech, Natural Language Processing, and others. A deep neural network needs to be trained from a large corpus of data (>>1TB), typically imagery, video, audio, or text. Such training procedures often require customization of the neural network architecture, learning criteria, and dataset pre-processing. In addition to the computational expense demanded by the learning algorithms, there is a high need for rapid prototyping and, thus, ease of development. Some of the largest applications currently are in image recognition and scientific studies with 10 million images and up to 11 billion parameters, for supervised and unsupervised learning. In future, applications such as training of self-driving cars may require processing of ~100 million images at megapixel resolution.

While these examples provide a glimpse of the vast potential of data-intensive approaches in real applications, they also illustrate the challenges in defining the scope of the benchmarking problem for big data applications. The next section provides two specific approaches to tackling this issue.

3 Levels of Abstraction

Since the scope of big data applications can be vast—as described in the previous section—it is important to develop "abstractions" of real applications, in order to then develop benchmark specifications, which are based on those abstractions. The two abstractions described in this section are based on (1) extending the familiar *data warehouse* model to include certain big data characteristic in the data and the workload and (2) specifying a *pipeline* of processing, where data is transformed and processed in several steps. The specifics of each step may be different for different applications domains.

As described in the Section 1, TPC benchmarks provide an application-level abstraction of business applications, such as transaction processing and/or complex query processing. TPC benchmarks, such as TPC-C, TPC-H, TPC-DS[4], model retail transaction processing and data warehousing environments. The database model and workload provide a representative view of a specific business application scenario. Nonetheless, the results of such benchmarks can be used as a guide for a variety of other application (non-business) use cases with similar characteristics. Following this example, we present two models for big data benchmarks, in this section. The first, *BigBench* follows the TPC model; while the second, *Deep Analytics Pipeline*, is a model based on a generalization of big data processing pipelines.

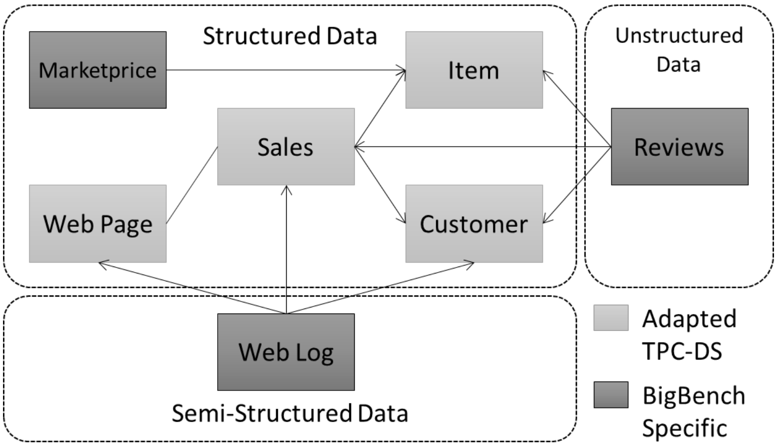

Fig. 1. BigBench Data Model

3.1 BigBench

BigBench is a big data analytics benchmark based on TPC-DS. Its development was initiated at the first *Workshop on Big Data Benchmarking* in May 2012 [5]. BigBench models a retail warehouse that has two sale channels: web sales and store sales. An

[4] TPC, http://www.tpc.org/

excerpt of the data model is shown in **Fig. 1**. In order to appropriately reflect the big data use case, BigBench features not only structured data but also semi-structured and unstructured data. These data sections contain dependencies. For example, the web log, or clickstream data (semi-structured), has references to the `SALES` table. There are thirty queries specified in the BigBench workload, which cover a broad variety of analytics representing the different big data levers that were identified by Manyika et al. [6]. The queries use cases cover business aspects of marketing, merchandising, operations, supply chains, and reporting in a typical enterprise.

The BigBench processing model is targeted at batch analytics. The complete benchmark process consisting of three stages: data generation, data load, a *Power Test*, and a *Throughput Test*. The data generation step generates the complete data set in different flat file formats. However, it is not part of the measured benchmark run. In the loading stage, data can be loaded into the system under test. The *Power Test* is performed by a serial execution of all thirty queries in the workload, while the *Throughput Test* consists of running a preselected number of serial streams, each of which is a permutation of the thirty queries. The total number of queries run is divided by the benchmark runtime to obtain a queries-per-hour metric.

BigBench has gained widespread attention in industry and academia and is currently in process to be standardized by the TPC. Several extensions were proposed [7].

3.2 Data Analytics Pipeline

Another proposal for a big data benchmark, referred to as the *Deep Analytics Pipeline,* is based on the pipeline of processing typically found in big data applications, from data ingestion to data analysis and use. **Fig. 2** shows the steps in such a pipeline [12].

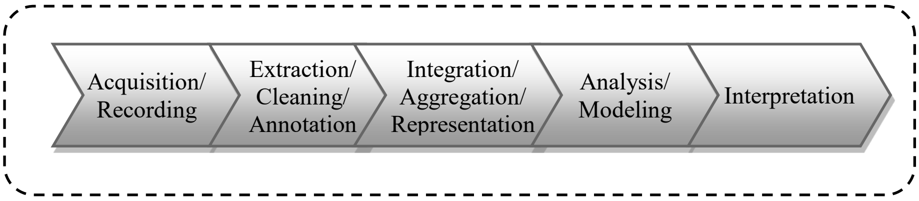
Acquisition/ Recording
Extraction/ Cleaning/ Annotation
Integration/ Aggregation/ Representation
Analysis/ Modeling
Interpretation

Fig. 2. Deep Analytics Pipeline

As described in [12], many data-driven industries are engaged in attempting to identify and learn the behavior of entities and events of interest. For example, the online advertising industry is attempting to learn *user activities* that are of consequence, i.e., activities that eventually lead to clicking on an online advertisement. The banking industry is interested in predicting *customer churn* based on the customer data, such as, say, demographics, income, and interaction patterns that are available to them. The insurance industry is interested in predicting *fraud* based on the data about their customers' activities, while the healthcare industry would like to predict a patient's propensity to visit the emergency room, and the *need for preventive care* based on patient data. All of these use cases involve collecting a variety of data sets about the entities of interest, and detecting correlations between the “interesting” outcomes and prior behavior. Thus, such "user modeling” pipelines provide a typical use case for a large class of big data applications.

Pipelined data processing is also very typical of many, if not most, scientific applications as well. The processing pipeline in this case includes steps of data acquisition, data ingestion (data cleaning and normalization), data validation, and a variety of downstream analytics and visualization. The pipeline model is, thus, generic in nature. Different classes of application may be characterized by variations in the steps of the pipeline.

The stage of the pipeline may be executed on a single platform, or distributed across different platforms. Each stage is described in terms of its functionality, rather than in platform-specific terms.

4 Benchmarks Standards

Current industry standards provide two successful models for data-related benchmarks, TPC and SPEC, both of which were formed in 1988. TPC was launched with the objectives of creating standard benchmarks and a standard process for reviewing and monitoring those benchmarks [8]. SPEC was founded in the same year by a small number of workstation vendors who realized that the marketplace was in desperate need of realistic, standardized performance tests. The key realization was that an “ounce of honest data was worth more than a pound of marketing hype.” Interestingly, the current community interest in big data benchmarking has the same motivation! Both organizations, TPC and SPEC, operate on a membership model. Industry as well as academic groups may join these organizations.

4.1 The TPC Model

TPC Benchmarks are free for download; utilize standardized metrics for measuring transaction and query throughput; measure performance as well as price performance of given solutions; and, have more recently introduced an energy metric, to measure performance versus energy consumption.

TPC benchmarks are designed to test the performance of the entire system—hardware as well as software, using metrics for transactions or queries per unit of time. The specifications are independent of the underlying hardware and software implementations. Over its history, TPC has demonstrated that its benchmarks have relatively long “shelf life”. Benchmarks remain valid for several years and, in the case of TPC-C, it is worth noting that the 22-year old benchmark is still valid! The benchmark measures transactions per minute for a scenario based on Order-Entry systems. The transactions include entering and delivering orders, recording payments, checking the status of orders, and monitoring the level of stock at warehouses. One of the keys to the longevity of the TPC-C benchmark is the rule for “data scaling”, which is based on a “continuous scaling” model, where the number of warehouses in the database scales up with the number of transactions.

TPC benchmark "sponsors" may publish official TPC results for a fee. The publication rules requires full disclosure of all information, including system configuration, pricing, and details of performance. Benchmark results are audited by an independent, third party auditor.

TPC benchmarks that are query processing-oriented specify fixed database sizes at which the benchmark may be executed. These, so-called database *scale factors* range from 1—representing a 1GB "raw" database size—to 100,000 for a 100TB raw database size. The most common scale factors for which benchmarks have been published are in the range from 100 to 10,000 (100GB to 10TB). Big data benchmarks may also need to adopt a similar scheme for database sizes.

4.2 The SPEC Model

SPEC benchmarks typically focus on specific functions or operations within a system, e.g. integer performance, sort performance. The benchmarks are typically server-centric, and test performance of small systems or components of systems. Unlike TPC, each benchmark defines its own metric, since different benchmarks may focus on different aspects of a system. As a result, they tend to have short shelf life—benchmarks can become obsolete with a new generation of hardware or system. The SPEC benchmark toolkits can be downloaded for a fee.

Publication of benchmark results by SPEC is free to members and subject to a modest fee for non-members. Unlike TPC, which incorporates third-part audits, SPEC benchmark results are peer reviewed. Also, unlike TPC, which requires extensive, full disclosure, SPEC requires only a disclosure summary—partly because the benchmark is at a component level within a system. The following table summarizes the features of TPC vs SPEC benchmarks.

TPC Model	**SPEC Model**
Specification based	Kit based
Performance, price, energy in one benchmark	Performance and energy in separate benchmarks
End-to-end benchmark	Server-centric benchmark
Multiple tests (ACID, load, etc.)	Single test
Independent review	Peer review
Full disclosure	Summary disclosure

4.3 Elasticity

An important requirement for big data systems is *elasticity.* For big data systems that deal with large data that are continuously growing, e.g., clicks streams and sensor streams, the system must be designed to automatically take advantage of additional resources, e.g., disks or nodes, as they are added to the system. Conversely, given the scale of many big data systems, there is a high probability of component failures during any reasonable workload run. The loss of a component in such a system should not lead to application failure, system shutdown, or other catastrophic results. The system should be “elastic” in how it also adopts to addition and/or loss of resources. Benchmarks designed for big data system should attempt to incorporate these features as part of the benchmark itself, since they occur as a matter of course in such systems. Currently, TPC benchmarks do require ACID tests (for testing Atomicity, Consistency, Isolation, and Durability properties of transaction processing and/or database systems). However, such tests are done outside the benchmark window i.e., they are not a part of the benchmark run itself, but are performed separately.

5 Current Benchmarking Efforts

Several efforts have been created to foster the development of big data benchmarks. A recent paper summarizes a number of such efforts [9]. The first *Workshop on Big Data Benchmarking*, which was held in San Jose, California, USA, in May 2012, created one of the first community forums on this topic. In this workshop, sixty participants from industry and academia came together to discuss the development of industry standard big data benchmarks. The workshop resulted in two publications [1, 2] and the creation of the big data Benchmarking Community (BDBC), an open discussion group that met in biweekly conference calls and via online discussions. In mid-2014, with the creation of the SPEC Research Group on big data[5], the BDBC group was merged with the SPEC activity. With the formation of the SPEC RG, weekly calls have been established, as part of the SPEC activity, with the weekly presentations alternating between open and internal calls. The open presentations cover new big data systems, benchmarking efforts, use cases, and related research. The internal calls are restricted to SPEC members only, and focus on discussion of big data benchmark standardization activities.

The WBDB series launched in 2012 has been continuing successfully, with workshops in the India (December 2012), China (July 2013), US (October 2013), Germany (October 2014), and Canada (June 2015). The next workshop, the 7th WBDB, will be held on December 14-15 in New Delhi, India. The WBDB workshops have, from the beginning, included participation by members of standards bodies, such as SPEC and TPC. Workshop discussions and papers presented at WBDB have led to the creation of the SPEC Research Group on big data, as mentioned earlier, and creation of the TPC Express Benchmark for Hadoop Systems (*aka* TPCx-HS), which is the first industry standard benchmark for Apache Hadoop compatible big data systems, and is based on the Terasort benchmark. The BigBench paper presented first at WBDB has also led to the formation of the TPC-BigBench subcommittee, which is working towards a big data benchmark based on a data warehouse-style workload.

Finally, an idea that has been discussed at WBDB is the notion of creating a BigData Top100 List[6], based on the well-known TOP500 list used for supercomputer systems. Similar to the TOP500, the BigData Top100 would rank the world’s fastest big data

[5] SPEC RG big data - http://research.spec.org/working-groups/big-data-working-group

[6] BigData Top100 - http://www.bigdatatop100.org/

systems—with an important caveat. Unlike the TOP500 list, the BigData Top100 List would include a price/performance metric.

6 Conclusion

Big data benchmarking has become an important topic of discussion, since the launching of the WBDB workshop series in May 2012. A variety of projects in academia as well as industry are working on this issue. The WBDB workshops have provided a forum for discussing the variety and complexity of big data benchmarking—including discussions of who should define the benchmarks, e.g., technology vendors versus technology users/customers; what new features should be include in such benchmarks, that have not been considered in previous performance benchmarks, e.g., *elasticity* and *fault tolerance*.

Even though this is a challenging topic, the strong community interest in developing standards in this area has resulted in the creation of the TPCx-HS benchmark; formation of the TPC-BigBench subcommittee; and, the formation of the SPEC Research Group on big data. Finally, a benchmark is only a formal, standardized representation of "typical" real-world workloads that allows for comparability among different systems. Eventually, users are interested in the performance of their specific workload(s) on a given system. If a given workload can be formally characterized, it could then be executed as a service across many different systems, to measure the performance of any system on that workload.

7 References

1. Baru, C., Bhandarkar, M., Poess, M., Nambiar, R., Rabl, T., Setting the Direction for big data Benchmark Standards, TPC-Technical Conference, VLDB 2012, July 26-28, 2012, Istanbul, Turkey.
2. 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, March 2013.
3. Tilmann Rabl, Meikel Poess, Chaitan Baru, and Hans-Arno Jacobsen. Specifying big data Benchmarks. Volume 8163 of LNCS. Springer Berlin Heidelberg, 2014.
4. Tilmann Rabl, Raghunath Nambiar, Meikel Poess, Milind Bhandarkar, Hans-Arno Jacobsen, and Chaitan Baru. Advancing big data Benchmarks. Volume 8585 of LNCS. Springer Berlin Heidelberg, 2014.
5. Ghazal, A., Rabl, T., Hu, M., Raab, F., Poess, M., Crolotte, A., Jacobsen., H.A. BigBench: Towards an industry standard benchmark for big data analytics, Proceedings of the 2013 ACM SIGMOD Conference.
6. Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., Byers, A.H. Big data: The Next Frontier for Innovation, Competition, and Productivity. Technical report, McKinsey Global Institute. 2011. http://www.mckinsey.com/insights/mgi/research/technology_and_innovation/big_data_the_next_frontier_for_innovation.
7. Chaitanya Baru, Milind Bhandarkar, Carlo Curino, Manuel Danisch, Michael Frank, Bhaskar Gowda, Hans-Arno Jacobsen, Huang Jie, Dileep Kumar, Raghunath Nambiar,

Meikel Poess, Francois Raab, Tilmann Rabl, Nishkam Ravi, Kai Sachs, Saptak Sen, Lan Yi, and Choonhan Youn. Discussion of BigBench: A Proposed Industry Standard Performance Benchmark for big data. TPC-Technical Conference, VLDB 2014.

8. Kim Shanley, History and Overview of the TPC, February, 1998, http://www.tpc.org/information/about/history.asp.
9. Todor Ivanov, Tilmann Rabl, Meikel Poess, Anna Queralt, John Poelman and Nicolas Poggi, Big Data Benchmark Compendium, TPC Technical Conference, VLDB 2015, Waikoloa, Hawaii, Aug 31, 2015.
10. Anon et al., A Measure of Transaction Processing Power, Datamation, 1 April, 1985.
11. Omri Serlin, The History of DebitCredit and the TPC, http://research.microsoft.com/en-us/um/people/gray/benchmarkhandbook/chapter2.pdf.
12. Chaitanya Baru, Milind Bhandarkar, Raghunath Nambiar, Meikel Poess, Tilmann Rabl, Benchmarking Big Data Systems and the BigData Top100 List, Big Data. March 2013, 1(1): 60-64.