ISSN No. 0976-5697

Volume 5, No. 8, Nov-Dec 2014

International Journal of Advanced Research in Computer Science

REVIEW ARTICLE

Available Online at www.ijarcs.info

Overview on Performance Testing Approach in Big Data

Ashlesha S. Nagdive
Department of Information Technology
G.H Raisoni College of Engineering
Nagpur, India

Dr. R. M. Tugnayat
Professor and Principal
Shri Shankarprasad Agnihotri College of Engineering
Wardha, India

Manish P. Tembhurkar
Department of Computer Science & Engineering
G.H Raisoni College of Engineering
Nagpur, India

Abstract: Big data is defined as large amount of data which requires new technologies and architectures so that it becomes possible to extract value from it by capturing and analysis process. Big data due to its various properties like volume, velocity, variety, variability, value, complexity and performance put forward many challenges. Many organizations are facing challenges in facing test strategies for structured and unstructured data validation, setting up optimal test environment, working with non relational database and performing non functional testing.

These challenges cause poor quality of data in production, delay in implementation and increase in cost. Map Reduce provides a parallel and scalable programming model for data-intensive business and scientific applications. To obtain the actual performance of big data applications, such as response time, maximum online user data capacity size, and a certain maximum processing capacity.

Keywords: Bigdata, Testing strategies, MapReduce, Hadoop, performance testing

I. INTRODUCTION

Big data is an all-encompassing term for any collection of data sets so large and complex that it becomes difficult to process them using traditional data processing applications. Big data usually includes data sets with sizes beyond the ability of commonly used software tools to capture, curate, manage, and process data within a tolerable elapsed time [1]. Big data "size" is a constantly moving target, as of 2012 ranging from a few dozen terabytes to many petabytes of data. Big data is a set of techniques and technologies that require new forms of integration to uncover large hidden values from large datasets that are diverse, complex, and of a massive scale. Big data uses inductive statistics and concepts from nonlinear system identification to infer laws from large sets of data with low information density to reveal relationships, dependencies and perform predictions of outcomes and behaviors[2].

Due to such large size of data it becomes very difficult to perform effective analysis using the existing traditional techniques. Big data due to its various properties like volume, velocity, variety, variability, value, complexity and performance put forward many challenges.[3]

Testing Big data is one of biggest challenge faced by every organization because of lack of knowledge on what to test and how to test. Biggest challenges faced in defining test strategies for structured and unstructured data validation, setting up an optimal test environment, working with non relational database and performing non –functional testing. These challenges cause poor quality of data in production and delayed implementation and increase in cost [4].

The big data application will handle a large number of structured and unstructured data. The data processing will involve more than one data node and completed in a shorter period of time. Due to the low quality and poor system design code, application performance as data volume growth will decline, even when the amount of data reaches a certain size, the application crashes and cannot provide mission services. If the performance of the application does not meet the service level agreements (Service-Level Agreement, SLA), will lose the goal of building big data systems. Therefore, due to data capacity size and complexity of systems in big application, performance testing has played a very important role to achieve the actual performance ability [4].

II. LITERATURE REVIEW

Given its current popularity, the definition of big data is rather diverse, and reaching a consensus is difficult. Fundamentally, big data means not only a large volume of data but also other features that differentiate it from the concepts of “massive data" and “very large data". In fact, several definitions for big data are found in the literature, and three types of definitions play an important role in shaping how big data is viewed:

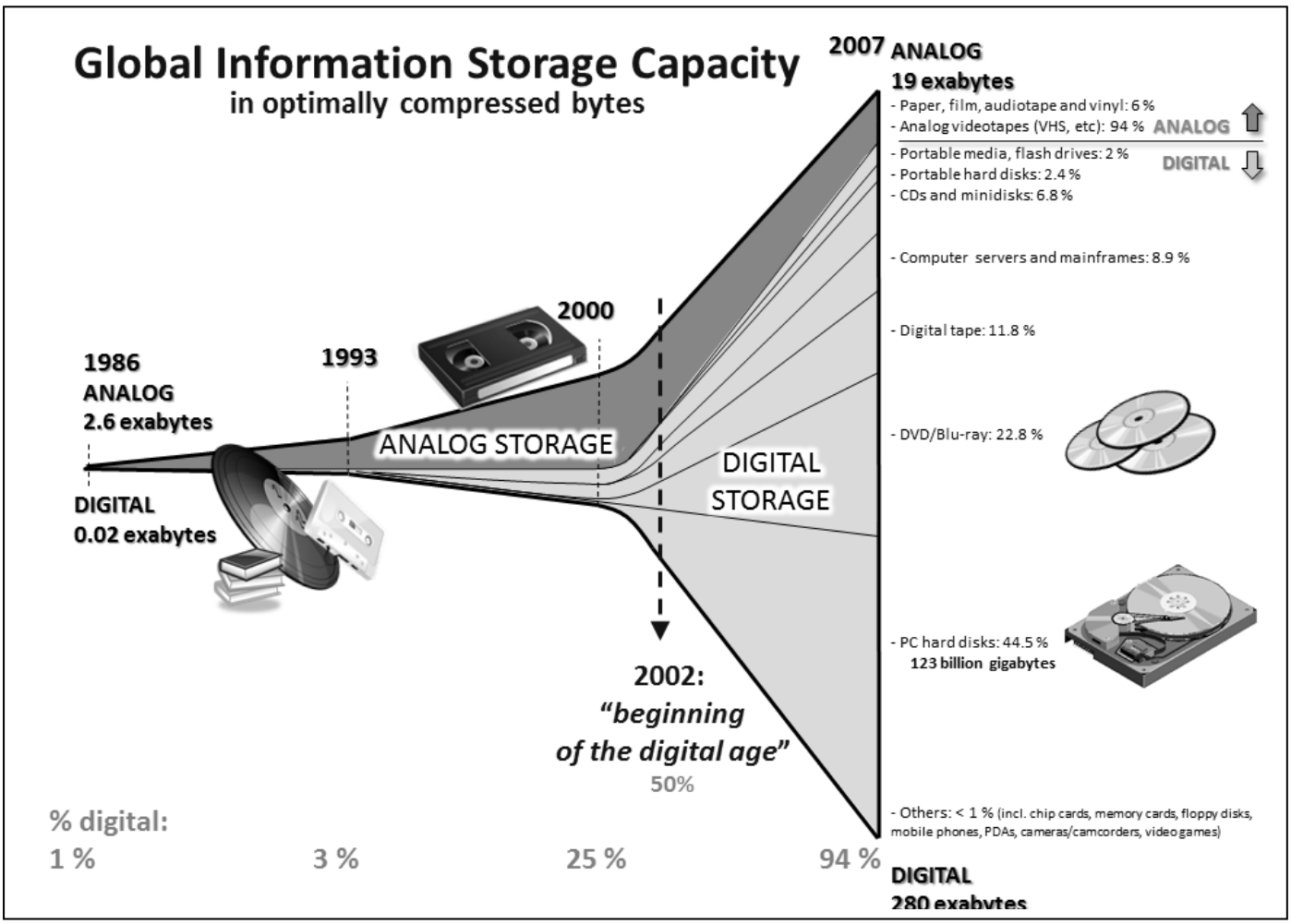

Figure 1: Growth of and Digitization of Global Information Storage Capacity [12]

A. Attributive Definition:

IDC is a pioneer in studying big data and its impact. It defines big data in a 2011 report that was sponsored by EMC (the cloud computing leader) : "Big data technologies describe a new generation of technologies and architectures, designed to economically extract value from very large volumes of a wide variety of data, by enabling high-velocity capture, discovery, and/or analysis." This definition delineates the four salient features of big data, i.e., volume, variety, velocity and value. As a result, the ``4Vs" definition has been used widely to characterize big data. A similar description appeared in a 2001 research report in which META group (now Gartner) analyst Doug Laney noted that data growth challenges and opportunities are three-dimensional, i.e., increasing volume, velocity, and variety. Although this description was not meant originally to define big data, Gartner and much of the industry, including IBM and certain Microsoft researchers, continue to use this ``3Vs" model to describe big data 10 years later [5].

B. Comparative Definition:

In 2011, Mckinsey's report defined *big data* as "datasets whose size is beyond the ability of typical database software tools to capture, store, manage, and analyze." This definition is subjective and does not define big data in terms of any particular metric [6]. However, it incorporates an evolutionary aspect in the definition (over time or across sectors) of what a dataset must be to be considered as big data.

C. Architectural Definition:

The National Institute of Standards and Technology (NIST) suggests that, "Big data is where the data volume, acquisition velocity, or data representation limits the ability to perform effective analysis using traditional relational approaches or requires the use of significant horizontal scaling for efficient processing" In particular, big data can be further categorized into big data science and big data frameworks. Big data science is "the study of techniques covering the acquisition, conditioning, and evaluation of big data," whereas big data frameworks are "software libraries along with their associated algorithms that enable distributed processing and analysis of big data problems across clusters of computer units". An instantiation of one or more big data frameworks is known as big data infrastructure [5].

III. TESTING STRATEGIES

Different testing types like functional and non functional testing are required along with strong test data and test environment management to ensure that the data from varied sources is processed error free and can obtained good quality to perform analysis. Functional testing activities like validation of map reduce process, structured and unstructured

data validation, data storage validation are important to ensure the data is correct and is of good quality[3].

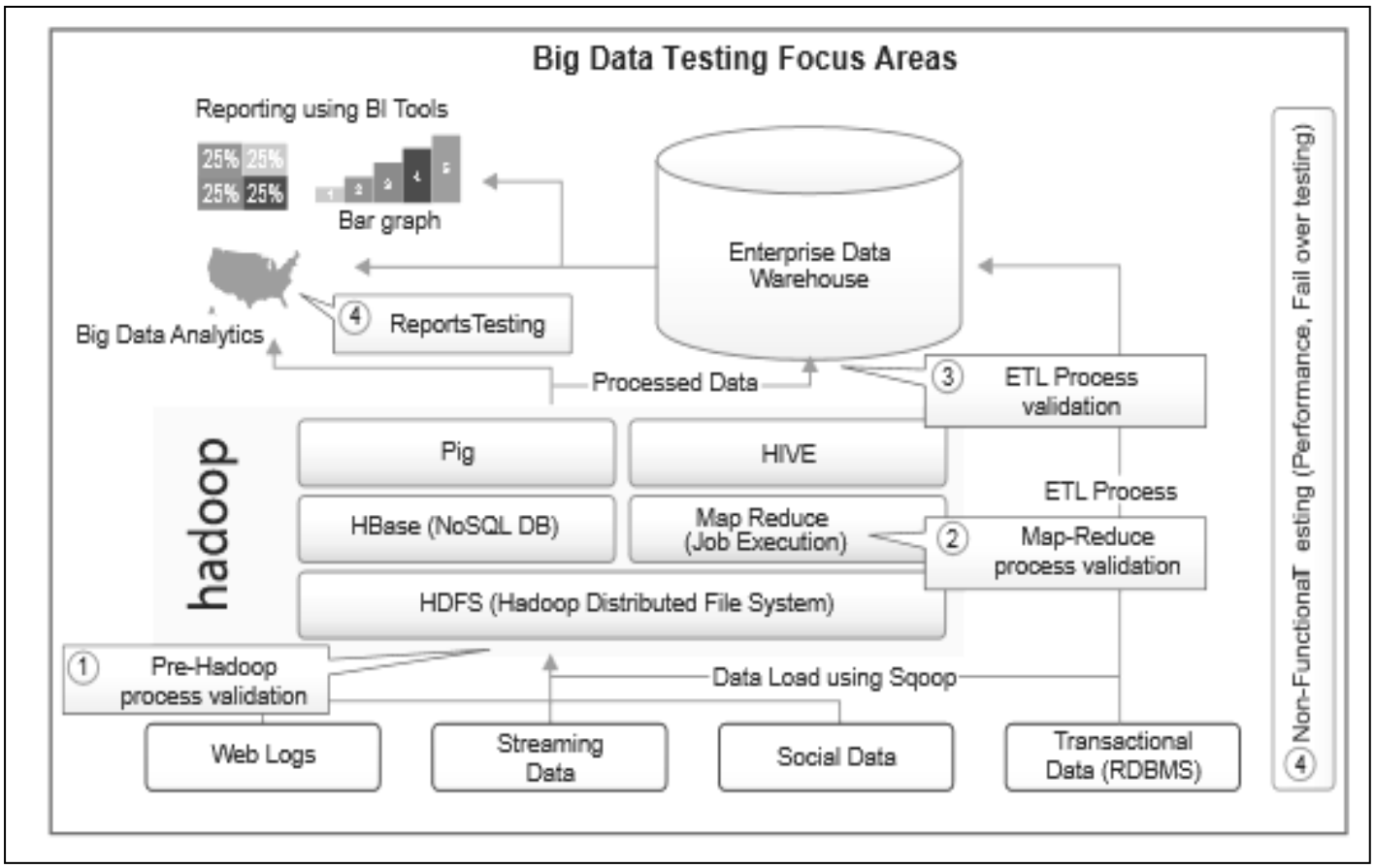

Figure 2: Big data architecture [3]

Hadoop is a framework that allows for distributed processing of large data sets across clusters of computer. Hadoop uses Map/reduce, where the application is divided into many small fragments of work, which may be executed on any node in the cluster [3]. The process is illustrated below by an example based on the open source Apache Hadoop software framework:

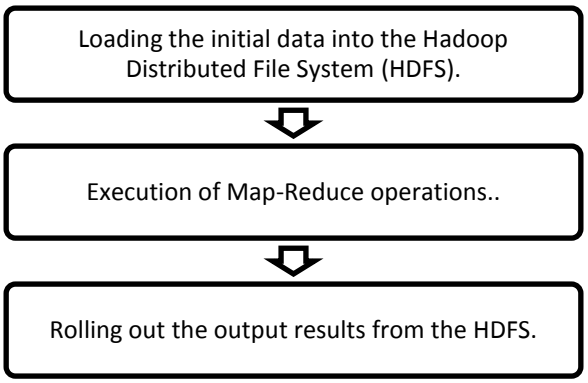

Figure 3: Process Flowchart of Big data Framework

A. Loading the initial data into the Hadoop Distributed File System (HDFS).

B. Execution of Map-Reduce operations..

C. Rolling out the output results from the HDFS.

A. Loading the Initial Data into HDFS

In this first step, the data is retrieved from various sources (social media, web logs, social networks etc.) and uploaded into the HDFS, being split into multiple files:

- Verify that the required data was extracted from the original system and there was no data corruption [4].
- Validate that the data files were loaded into the HDFS correctly [4].
- Check the files partition and copy them to different data units [4].
- Determine the most complete set of data that needs to be checked [4].

B. Execution of Map-Reduce Operations[4]

In this step, you process the initial data using a Map-Reduce operation to obtain the desired result. Map-reduce is a data processing concept for condensing large volumes of data into useful aggregated results:

- Check required business logic on standalone unit and then on the set of units.
- Validate the Map-Reduce process to ensure that the "key-value" pair is generated correctly.
- Check the aggregation and consolidation of data after performing "reduce" operation.

- Compare the output data with initial files to make sure that the output file was generated and its format meets all the requirements.

C. Rolling out the Output Results from HDFS:

This final step includes unloading the data that was generated by the second step and loading it into the downstream system, which may be a repository for data to generate reports or a transactional analysis system for further processing: Conduct inspection of data aggregation to make sure that the data has been loaded into the required system and thus was not distorted. Validate that the reports include all the required data and all indicators are referred to concrete measures and displayed correctly [4].

As speed is one of Big Data's main characteristics, it is mandatory to do performance testing. A huge volume of data and an infrastructure similar to the production infrastructure is usually created for performance testing. Furthermore, if this is acceptable, data is copied directly from production. To determine the performance metrics and to detect errors, you can use, for instance, the Hadoop performance monitoring tool. There are fixed indicators like operating time, capacity and system-level metrics like memory usage within performance testing[4].To be successful, Big Data testers have to learn the components of the Big Data ecosystem from scratch. Since the market has created fully automated testing tools for Big Data validation, the tester has no other option but to acquire the same skill set as the Big Data developer in the context of leveraging the Big Data technologies like Hadoop. This requires a tremendous mindset shift for both the testers as well as testing units within organizations. In order to be competitive, companies should invest in Big Data-specific training needs and developing the automation solutions for Big Data validation [4].

IV. TOOLS AND TECHNIQUES AVAILABLE

A. Hadoop:

Hadoop is an open source project hosted by Apache Software Foundation, a framework for distributed storage and distributed processing of Big Data on clusters of commodity hardware. Its Hadoop Distributed File System (HDFS) splits files into large blocks (default 64MB or 128MB) and distributes the blocks amongst the nodes in the cluster [5]. For processing the data, the Hadoop Map/Reduce ships code to the nodes that have the required data and the nodes then process the data in parallel. This approach takes advantage of data locality, in contrast to conventional HPC architecture which usually relies on a parallel file system. It consists of many small sub projects which belong to the category of infrastructure for distributed computing. Hadoop mainly consists of [5]:

- File System (The Hadoop File System)
- Programming Paradigm (Map Reduce)

There are various problems in dealing with storage of large amount of data. Though the storage capacities of the drives have increased massively but the rate of reading data from them hasn't shown that considerable improvement. There occur many problems also with using many pieces of hardware as it increases the chances of failure. This can be avoided by Replication i.e. creating redundant copies of the same data at different devices so that in case of failure the copy of the data is available. The main problem is of combining the data being read from different devices. Many a methods are available in distributed computing to handle this problem but still it is quite challenging. Such problems are easily handled by Hadoop [5]. The problem of failure is handled by the Hadoop Distributed File System and problem of combining data is handled by Map reduce programming Paradigm. Map Reduce reduces the problem of disk reads and writes by providing a programming model dealing in computation with keys and values. Hadoop thus provides: a reliable shared storage and an analysis system. The storage is provided by HDFS and analysis by MapReduce.

B. MapReduce:

MapReduce is the programming paradigm allowing massive scalability. The MapReduce basically performs two different tasks i.e. Map Task and Reduce Task [5]. A map-reduce computation executes as follows: Map tasks are given input from distributed file system. The map tasks produce a sequence of key-value pairs from the input and this is done according to the code written for map function. These value generated are collected by master controller and are sorted by key and divided among reduce tasks [5][6]. The sorting basically assures that the same key values ends with the same reduce tasks. The Reduce tasks combine all the values associated with a key working with one key at a time. Again the combination process depends on the code written for reduce job. The Master controller process and some number of worker processes at different compute nodes are forked by the user. The Master controller creates some number of maps and reduces tasks which are usually decided by the user program. The tasks are assigned to the worker nodes by the master controller. Track of the status of each Map and Reduce task is kept by the Master Process. The failure of a compute node is detected by the master as it periodically pings the worker nodes. All the Map tasks assigned to that node are restarted even if it had completed and this is due to the fact that the results of that computation would be available on that node only for the reduce tasks. The status of each of these Map tasks is set to idle by Master. These get scheduled by Master on a Worker only when one becomes available. The Master must also inform each Reduce task that the location of its input from that Map task has changed [5].

V. NON FUNCTIONAL TESTING

A. Performance Testing

Big data applications through performance testing, we can achieve the following objectives

1) Obtain the actual performance of big data applications, such as response time, maximum online user data capacity size, and a certain maximum processing capacity.
2) Access performance limits and found that the conditions can cause performance problems, such as testing under load is applied to some problems can occur after a long run in big data application.
3) Achieve performance status and resource status in big data application, and to optimize the performance parameters in big data applications (eg. hardware

configuration, parameter configuration and application-level code).

The purposes of performance testing are not only acknowledging application performance levels to, but to improve the performance of the big data application. Before performance testing, test engineers should fully consider their testing requirement and then design a complete test scenario to consider the test program with the actual situation of the user operation [10]. Through the test execution and results analysis, performance bottlenecks can be found and analysis the reason further. In the performance test, test engineers need to collect the resource use information during performance test execution. Related to response time, the collecting resources use information, the more obtained performance information analysis, and the more analysis of system performance bottleneck[11].Not only for big data application infrastructure, data processing capabilities, network transmission capacity in depth testing, but also from the basic characteristics of big data to analyze the factors affecting the performance of big data applications. In big data applications, the rapid growth of mobile computing and network users, mobile devices, changing only the type of data occurs, and the data generated is very rapid with increase of real-time data transactions [11].

If the performance does not meet SLA, the purpose of setting up Hadoop and other big data technologies fails. Hence the performance testing plays vital role in big data project.

B. FailoverTesting:

Hadoop architecture consists of node and hundreds of data nodes hosted on server machine .There are chances of node failure and HDFS components become non functional. HDFS architecture is designed to detect these failures and automatically recover to proceed with the processing. Failover testing validates the recovery process and ensures the data processing when switched to other data nodes. Recovery Time Objective (RTO) and Recovery Point Objective (RPO) metrics are captured during failover testing.

VI. CONCLUSION

This paper described the overview of problems faced by Big data storage and inconsistency. The challenge faced today is how to test big data and improving the performance of the big data application [6]. Hadoop tool for Big data is described in detail. Map Reduce provides a parallel and scalable programming model for data-intensive business and scientific applications [7]. Various testing strategies are studied required for big data. We propose a performance diagnostic methodology that integrates statistical analysis from different layers, and design a heuristic performance diagnostic tool which evaluates the validity and correctness of Hadoop by analyzing the job traces of popular big data benchmarks[8][9]. We can obtain the actual performance of big data applications, such as response time, maximum online user data capacity size, and a certain maximum processing capacity. The technology provided test goal analysis, test design, load design for big data application [10], [11].

VII. REFERENCES

[1] Zhenyu Liu, *"Research of Performance Test Technology for Big Data Applications",* in IEEE International Conference on Information and Automation Hailar, China, July 2014.

[2] Jie Li,, Zheng Xu, , Yayun Jiang and Rui Zhang, "*The Overview of Big Data Storage and Management",* Proc. 2014 IEEE 13th Int'l Conf. on Cognitive Informatics & Cognitive Computing (ICCI'CC'14), 2014.

[3] Roberto Paulo Andrioli de Araujo, Marcos Lordello Chaim, "*Data-flow Testing in the Large" ,* IEEE International Conference on Software Testing, Verification, and Validation 2014.

[4] Mahesh Gudipati,Shanthi Rao, Naju D. Mohan and Naveen Kumar Gajja, *"Big Data: Testing Approach to Overcome Quality Challenges*",Infosys labsBriefingVol 11, NO 1, 2013.

[5] Avita Katal, Mohammad Wazid, R H Goudar, *"Big Data: Issues, Challenges, Tools and Good Practices",* IEEE, 2013.

[6] Xiaoming Gao, Judy Qiu, "*Supporting Queries and Analyses of Large-Scale Social Media Data with Customizable and Scalable Indexing Techniques over NoSQL Databases*", 14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2014.

[7] Rongxing Lu, Hui Zhu, Ximeng Liu, Joseph K. Liu, and Jun Shao**, "***Toward Efficient and Privacy-Preserving Computing in Big Data Era",* IEEE Network, July/August 2014.

[8] Marcelo Veiga Neves, C´esar A.F. De Rose, Kostas Katrinis, Hubertus Franke, "*Pythia: Faster Big Data in Motion through Predictive Software-Defined Network Optimization at Runtime",* IEEE 28th International Parallel & Distributed Processing Symposium, 2014.

[9] Rosangela de F´atima Pereira, Walter Akio Goya, "*Exploiting Hadoop Topology in Virtualized Environments*", IEEE 10th World Congress on Services, 2014.

[10] Zibin Zheng, Jieming Zhu, and Michael R. Lyu, *"Service-generated Big Data and Big Data-as-a-Service: An Overview",* in IEEE International Congress on Big Data, 2013.

[11] Jun Fan, "*Diagnosing Virtualized Hadoop Performance from Benchmark Results: An Exploratory Study"* IEEE International Congress on Big Data, 2014.

[12] Martin Hilbert and Priscila López, "*The World's Technological Capacity to Store, Communicate, and Compute Information*", Science 2011 Publication, Volume 332, ino. 6025; p.60-65, 2011.

http://www.sciencemag.org/content/332/6025/60.full.pdf?keytype=ref&siteid=sci&ijkey=89mdkEW.yhHlM