# Software testing under hadoop environment: a review

C3 Assignment

Mit2019007

# Software testing under Hadoop environment: a review

**Hemant Kumar Lader***
*Indian Institute of Information Technology*
*Prayagraj, UP, India*

## ABSTRACT

Software testing is a critical and important task in the development of successful software and the lifecycle of a software. It defines the fineness of the system and confirms that the system is error free and free from defects and faults and fits in the belief and requirement elicitation by stakeholders and users. Testing a software which deals with big data is not an easy task as the data pool is huge and it is difficult to apply the normal testing approach to test such applications. Nowadays the application of Big data systems is increasing because of its high performing analytical features and handling huge data. So to manage product quality and process performance of the Big Data system is very important. This can be achieved by testing the application under the big data environment only. Such an environment is a Hadoop environment. Provides Hadoop distributed file system for big data applications. So this paper gives an overview of performing software testing is in a big data environment.

**Keywords:** Big Data System, Software Testing, Hadoop Environment

## 1. INTRODUCTION

Many applications are present today for processing a large volume of data to get the desired result from it, and many different testing tools are present to test big data-based software. But there is no such tool which explains how information is validated before and after processing. Testing can be done to make sure the quality of data so that cost and time can be

saved. Nowadays the application of Big data systems is increasing because of its high performing analytical features and handling huge data. So to manage product quality and process performance of the Big Data system is very important.

Hadoop is the implementation of Map-Reduce [2] to analyze huge volume of data that are distributed over many machines[1]. It is fault tolerance. It provides Map-Reduce implementation to inspect large data that are distributed over many machines. It is a fault tolerance system. Hadoop is Apache's open source project deployed on Hadoop core, that combines Hadoop Distributed File System and Map-reduce Technology. This HDFS file system allows storing a huge volume of files stored in a distributed cluster of machines. Software developed to be operable in this environment must be well tested and should be able to perform without faults. In [3], author uses Natural Language Processing to generate test cases for generating test cases to test for the application in Hadoop environment. In [4] Yusuke el al. proposes their approach for automatic testing of such applications. As these applications are way too difficult to test by creating manual test cases. In this review paper, some of the approaches to test the software in a Hadoop environment and its challenges are discussed.

## 2. HADOOP ENVIRONMENT

Hadoop is an open-source framework designed by Apache to work with big data usage and functionality. It allows distributed processing of huge data sets from various clusters of systems. It uses Map-reduce functionality. In Map-reduce the work is divided in small fragments and is sent to different nodes of the cluster to process it. Hadoop-based cluster construction methods include a single configuration method, a virtual distributed method, and a fully distributed method. First, the single configuration method runs Hadoop as a Java process on a local system in a non-distributed mode, which is mainly useful for debugging Hadoop-based applications. The virtual distributed method is to configure each name node and data node as virtual Java processes on one node and execute them. Finally, the fully distributed method is a

method of constructing a cluster with multiple nodes communicating through the TCP / IP protocol. In this paper, we build a Hadoop-based cluster in a fully distributed manner [5].

There are many issues in dealing with storage capacity of data of large volumes. Although storage drives have increased the rate of fetching data from these devices has not shown that improvement. This system needs better maintainability. There may occur many problems as many hardware are connected increases the chance of failure. This can be avoided by making redundant copies in different devices so in cse of failure the data is available at another device. Combining the distributed data from different machines for processing is the main problem. Though many ways of handling this issue are available but yet it is a challenging part.

## 3. TESTING STRATEGIES FOR HADOOP ENVIRONMENT

Mainly testing of software testing has two categories:

- Manual Testing
- Automated Testing

Testing performed by using tools that automates the written script and reduces effort and time using many automation tools is known as Automation testing. Many such tool provide record and play back features to automate recorded tests and generates output each time for different inputs in each iteration.

### 3.1. AUTOMATED TESTING

In [4], the author applied QuickCheck testing tool. Which is is an automatic testing tool built for Haskell programs. The main function of tool quickCheck checks if the parameters hold for randomly generated test cases when parameters are passed as its arguments.

In [3], Priyanka et al. proposed an automatic test case generation based on natural language processing in the Hadoop system. It generates test cases from the software requirement

document written in natural language. This solves the error created by humans while testing manually and ensures the coverage of requirements during requirement elicitation. It generates test cases early in the software development Lifecycle. Thy took requirement specification as natural language and processed using their proposed method by natural language processing. Then the method constructs a requirement isn a form of a tree. Used to generate knowledge graphs that can be traversed to do boundary value analysis etc.

### 3.2. DEPENDABILITY TESTING FRAMEWORK

Dependability testing of the application in a Hadoop environment requires validating the behavior of stimulation of application to its tolerated faults, that in turn requires external control over all the components of application and their processing steps [6]. Fault case is a set of components needed to complete the execution of a system and validating the dependability of a system.

## 4. COST OF TESTING

Software testing has an important role in improving the confidence for correctness, quality, and robustness of the software. Specifications share problems as software testing, like high cost and long running time.

As the size and complexity of software increase, its test cases and suite keep on extending and its execution time becomes an issue to software development. Many approaches are used to cut down the cost of time consuming test phases. One way is from the existing test suite selecting a representative subset, that reduces the cost of testing.

Implementing these techniques provide better product quality, this will enhance the fluid execution of the system.

## 5. CONCLUSION

The most challenging task for a tester is to maintain the pace along with the changing trend and dynamics of the technology. The tester is not required to know the underlying technical details of the scene but here this is testing Big Data Technology like in the Hadoop environment is different. Some strategies discussed in this paper can be useful for testing the software application which deals with tons of data, as testing manually will cost a lot and also time consuming, managing the testing phase is difficult for such an application. Test strategies like HadoopTest is a framework based on the Dependability test discussed in [6] automatically executes fault cases in real deployment scenarios.

## 6. REFERENCES


1. "The Apache Hadoop." *Http://Hadoop.apache.org/*, 2012, http://hadoop.apache.org/.
2. Dean, Jeffrey, and Sanjay Ghemawat. "MapReduce." *Communications of the ACM*, vol. 51, no. 1, 2008, pp. 107–113., doi:10.1145/1327452.1327492.
3. Priyanka Kulkarni and Yashada Joglekar, "Generating and Analyzing Test cases from Software Requirements using NLP and Hadoop ", International Journal of Current Engineering and Technology (INPRESSCO). 2014.
4. Wada, Yusuke, and Shigeru Kusakabe. "Performance Evaluation of A Testing Framework Using QuickCheck and Hadoop." Journal of Information Processing, vol. 20, no. 2, 2012, pp. 340–346., doi:10.2197/ipsjjip.20.340.
5. Marynowski, Joao Eugenio, and Andrey Ricardo Pimentel. "HadoopTest: A Dependability Testing Framework For Hadoop." Federal University of Parana´ Informatics Department, 2013.