**SparkFaultBench**

**Requirements Document**

**2016.9**

Contents

[SparkFaultBench Requirments Design 2](#_Toc26957)

[1. Background 2](#_Toc28530)

[2. Comparisions 3](#_Toc23252)

[3. Our work 4](#_Toc6319)

[4. Significance 5](#_Toc13001)

SparkFaultBench Requirements Design

# 1. Background

With the development of the Internet, mobile Internet and Internet of things, no one can deny that we have actually lived in an era of huge amounts of data. Analysis of these huge amounts of data has become a very import and urgent needs. But how to store, analyze and compute, and which platform can be a better choice, it is a hot topics. Massive scale data has brought unprecedented challenges. So many communities come up with their platforms to process the huge data, such Hadoop, Spark. But we choose Spark platform as a research object.

Spark is faster than other distributed cluster, such as Hadoop. The main reason why it has fast effect is that it process computation in its memory in order to iterate repeatedly by decreasing the transmission time of the I/O operation. The goodness of Spark is not only its RDD mechanism, multilingual interface, but also it provides a computing framework, such as Spark SQL, Spark Streaming, MLlib and GraphX that has lightweight API. Spark becoming the new core of big data analysis has five reason and we introduce the detail below.

Spark applies mainly to iterative computing. In order to reduce the cost of reading the disk frequently, it store the intermediate result in the memory by RDD mechanism. And its lightweight computing framework consists of real-time stream processing, machine learning, SQL querying and graph operation. These framework support lightweight API to process data.

Spark make programming easier. It support for script language such as Python and Scala which make user who has basic knowledge of programs play easy. Thus companies can implement the projects in a short time.

Spark supports abundant operator so that it can process complex queries. Hadoop only has map and reduce operator, while Spark extend the collection classifying into two categories, transformation and action, which provides ample interface for every module in Spark.

Spark can be merged to Hadoop easily. Due to the widely use of Hadoop, Spark construct on Hadoop file system, hdfs, to store files.

Spark is a open-source framework, programmer can do secondary development based on business needs.

Although Spark has so many merits for us, we do not know the the problem of inner Spark and its running effect of every module. We describe a benchmark of Spark to user who want to run program on the platform. It consists of running result of different data size and different cluster scale. For example, the user can know the data size and dimension that the platform can support by reading our benchmark.

# 2. Comparisons

However, there is nearly no benchmark for Spark, particularly in testing algorithms in Spark MLlib, Spark SQL, Spark GraphX.

Big data benchmark by Databricks, the startup focused on commercializing the popular Apache Spark data-processing framework, just testing that Spark is faster than Hadoop MapReduce. Just like this, we get it from Spark official website.

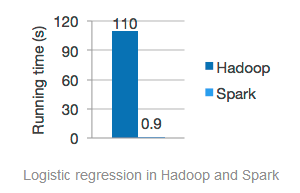


Figure 1 running results of Hadoop and Spark

If we just want to test the Spark, it has little help.

Spark Performance Tests from GitHub - databricks/spark-perf: Performance tests for Spark, test the performance of Spark MLlib and Spark Streamming. Not to say, it does not have Spark Sql and Spark Graphx tests. It is also difficult for users to use the tests. And we think that performance test is just a part of benchmark. It is not a integrated benchmark.

Spark SQL Performance Tests from GitHub - databricks/spark-sql-perf, test the performance of Spark SQL. Apparently, it does not have tests for other Spark algorithms. Besides, it can’t generate data and need to use an available dataset.

Besides the above benchmarks, you can also find other Spark benchmarks for a single aspect, such as Spark Sql benchmark, Spark MLlib benchmark and so on. But totally they all have following drawbacks.

1. No functions to generate data as you want.
2. No complete tests in Spark inner algorithm to find if it has problems.
3. No complete tests of performance in Spark inner algorithms.
4. Can’t choose parameters for benchmark.
5. No web interface so that it isn’t friendly to use

# 3. Our work

We purpose to design a benchmark tool based on spark system which can test each module performance,stability and scalability through generating different data.

Because of numbers of modules, we are not able to build testing for all spark modules. We decide to choose three typical modules which are Spark SQL, MLlib and GraphX to test. The whole procedure run as follow.

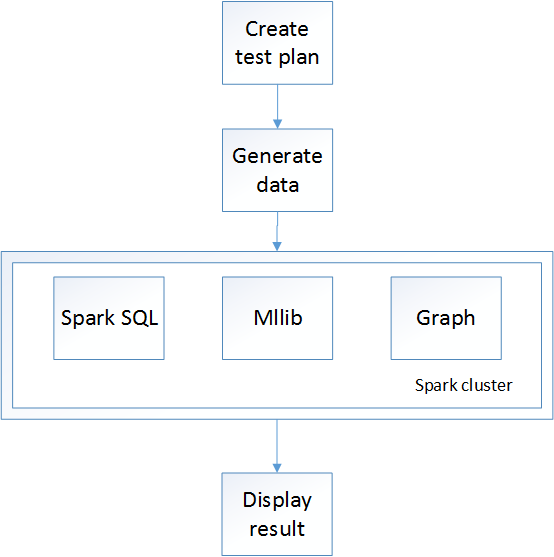


Figure 2 procedure chart

At first user create a testing, then system begin to generate data for testing, after that, start up spark cluster and run the module. Lastly, system display the testing results. According to above, we can get the functional requirements of the whole system:

1. Generate different test data

2. Build test programs for three modules

3. Display test results

# 4. Significance

With the Spark using widely nowadays, the requirement of the Spark performance and stability will be more important inevitable, which is not only a single module in spark. Our project exactly focus on these requirements. Through generating different data, the project tests the Spark system to show the integrate performance in different environment ,even to find out system bugs. In this way, our job has contributed to the development of spark system.