

# MapReduce and Apache Tez Data Compression Techniques

Chandrabhan S. Jadhao<sup>1</sup>, Prof. Harish K. Barapatre<sup>2</sup>

Department of Computer Engineering SES's Yadavrao Tasgaonkar Institute Of Engineering And Technology, Karjat,  
Maharashtra, India

## ABSTRACT

In Hadoop system, there are many challenges in dealing with enormous data sets. Regardless of whether you store your data, the fundamental challenge is that large volumes can usually cause network and INPUT/OUTPUT bottlenecks. One of the best data compression technique is using MapReduce, many instances of “map” steps process individual blocks of an inputs to produce one or more outputs; these outputs are passed to “reduce” steps where they are combined to produce a single result. MapReduce framework is a main engine of Hadoop cluster and widely used. It uses a batch oriented processing. Apache also developed an alternative engine called “Tez”. Which supports an interactive query and does not write temporary data into HDFS. This paper delves into MapReduce and Apache Tez data compression techniques that efficiently compresses and decompresses a large amount of data.

**Keywords:** Compression, Hadoop, HDFS, MapReduce, Tez.

## I. INTRODUCTION

Data can comprise structured and unstructured data; it exists in high and complex volumes and undergoes high rates of change. The key reason behind the grow of big data is its use to provide actionable insight. Typically, organizations use analytics applications to extract information that would otherwise be invisible, or impossible to derive using existing methods. Industries such as petrochemicals and financial services, large information servers, automobile industries, cab services have been using data warehousing techniques to process very large data for decades, but this is not what most understands as big data. The key difference is that today's big data sets include unstructured data and allow for extracting results from a variety of data types, such as emails, log files, social media, business transactions and a host of others. For example, sales figures of a chain of retail stores exist in a database and accessing them is not a big data problem. But, if the business wants to cross-reference sales of a particular item with weather conditions at time of sale, or with various customer

details, and to retrieve that information quickly, this would require intense processing and would be an application of big data technology, Hadoop is the answer for all big data solution. There are 2 frameworks to process data MapReduce framework and Tez framework. MapReduce framework supports batch processing and it is a default framework of Hadoop cluster. On the other hand, Tez supports interactive processing and it is complex to install and configuration with binary file from Apache Tez web site. Rupinder Singh's research invested Tez framework with Pig scripts and researchers purpose this framework that it is better for pre structure and data than MapReduce. This paper focuses on MapReduce and Tez data compression methods that are available on Hadoop cluster.

## II. BACKGROUND AND RELATED WORK:

### A. Related Works

Hadoop is the most popular issue in high performance Computing and big data now a days. In Rupinder Singh's Research, he focuses on a performance

comparison those of Hadoop's frameworks MapReduce and Tez with Pig scripts which is pre structure for data. The results of his research Show that Tez has better performance than MapReduce. In the Hadoop compression algorithms, Andre Wenas used GZIP, LZJB and ZLE Compression techniques for data Warehouse and his results show the best performance on ZLE. Yanpei Chen's research tries to select compress or decompress MapReduce output file for reduced power Consumption. His results shows decrease in energy Consumption more than 50%. Bhavin J. Mathiya use more Compression algorithm like DEFLATE, LZ4, Bzip and Gzip with word-count benchmark both with Map output and reduced output. He presented Bzip2 with higher compression ratio upto 85% than other techniques. This paper delve in to MapReduce and Apache Tez Data Compression Techniques.

### B. Hadoop

The most popular tools in a Big data solution is Hadoop. It is a portable software and widely used Commercially for example, OLA cabs, Google and yahoo. It's portable over the Java JDK and creates file system as Hadoop Distributed File System (HDFS) which support with a various operating system (e.g. Windows, Linux and Mac OS X). MapReduce and Tez are framework of Hadoop. In Hadoop Ecosystem, it has a lot of software built on top. For example Hive is relational warehouse, R Connectors for Statistic function with R language and Mahout is machine learning.

### C. Map-Reduce

Hadoop consists of two components Hadoop Distributes File System (HDFS) and Hadoop MapReduce which is used to store the big data and used to processes the data that is stored in HDFS respectively. The files stored in HDFS are divided into blocks, each block size is 128 MB by default. Every block is replicated onto a couple of data nodes, which by default are three. A Hadoop MapReduce task is divided into two steps Map phase and Reduce phase. Generally each map task processes a single HDFS block or multiple blocks. Once a map assignment finishes, its output is sent across the network to the nodes to reduce the tasks. Once every map task completes, the reduce phase starts. The reduce section calls reduce function and then stores the output to HDFS. Knowledge compression is used to reduce consumption of storage and INPUT/OUTPUT time. Essentially, compression reduces input/output time and increases compute time. Data compression can be used for map input, map output and reduce output data.. The compression ratio is calculated as a ratio of compressed data to uncompressed data. If input knowledge is compressed, a map task decompresses the information and then calls the user's map function. There are two types of compression codec available. (i) splittable formats, meaning if the input records is compressed with this form of code more than one task can manner the statistics in parallel. But if the codec is (ii) not splittable, then undertaking strategies complete input records. Non-splittable codec motive full size overall performance troubles while the dataset is large. excessive decompression and compression overhead.

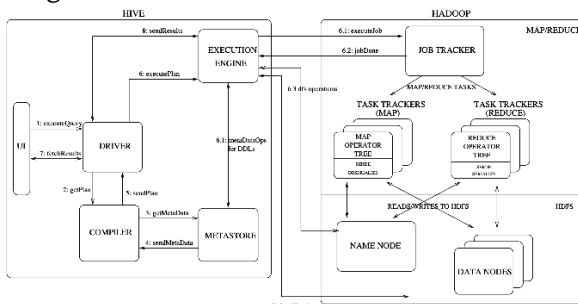


Figure 1. Hadoop eco-system

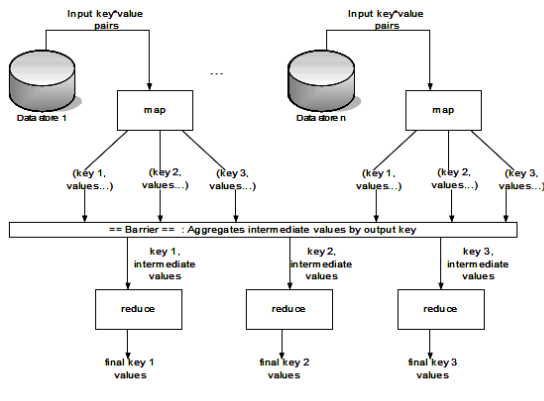


Figure 2. MapReduce Processing

**D. Tez**

Tez generalizes the MapReduce paradigm to a more powerful framework based on expressing computations as a dataflow graph. Tez is not meant for end-users. It just enables developers to build end-user applications with better performance and flexibility. Hadoop has traditionally been a batch-processing platform for large data. Tez is built on top of YARN, which is the new resource-management framework for Hadoop.

Tez is a next generation of distributed processing in Hadoop same as MapReduce. As MapReduce is not suitable for an interactive query, Tez is an interactive query and investigation for data structure on Hadoop. It is alternative for query processing. In Figure Tez processing model has only a single map phase and has multiple reduce phases. For the benefit of Tez, it does not require temporary storing data in HDFS during process that is significance with a performance.

**a)MapReduce Model**

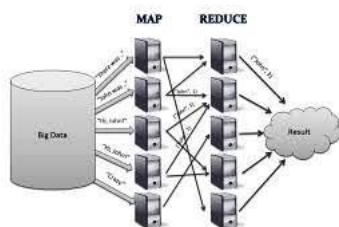


Figure 3. MapReduce Model

**b) Tez Model**

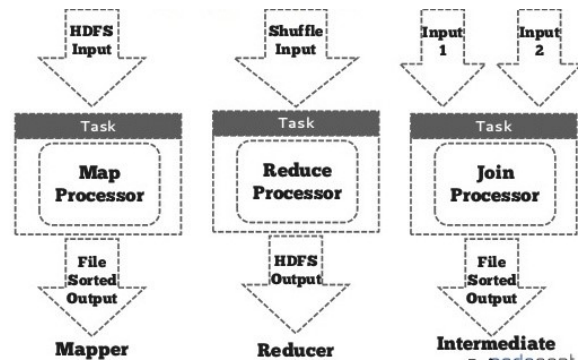


Figure 4. Runtime Tez Model

**E. Compression**

Linux operating system, has a native compression algorithm e.g. gzip, deflate and bzip2, gzip and deflate are the same general compression using zlib format. LZ777 concept uses Google based technology and design for Hadoop. On other side, bzip2 is only compression codec which splits file format.

**a) Bzip2:** Bzip2 is a high quality compression algorithm based on libbzip2. Which offers performance at twice time of compression and six times at decompression. It is useful for overfull disk drives, distribution CDs, backup taps and USB sticks. It reduces download times in network. In recovery mode, it can restore the compression data and decompress undamaged data.

**b) Snappy:** Snappy's previous name is Zippy. Developed in C++ by Google based with LZ77 concept, aims to increase the speed of compression. The benchmarks for snappy is used by Core i7 with a single core with 64-bits mode that results in compression ratio at 20-100% lower than gzip technic. In Hadoop cluster, snappy is most popular native compression technic that has widely used in, Hadoop, Cassandra LevelDB, MongoDB, Lucene RocksDB,.

**III. CONCLUSIONS**

A very large data size has some effects on a performance of reduce phase in Tez. In conclusion,

Hadoop cluster has become widely used with Apache Tez now a days as Tez framework has a better performance because it does not need to temporary store data in HDFS during process. As the result On the other hand, MapReduce always store data in HDFS that it has a direct effect of an execution time.

#### **IV. REFERENCES**

- [1]. Apache Software Foundation,
- [2]. R. Singh and P. J. Kaur, "Analyzing performance of Apache Tez and MapReduce" .
- [3]. A. wenas, S. Suharjito, "Improving Data Technology Warehouse Performance Using Filesystem with GZIP, LZJB and ZLE Compression".
- [4]. Y. Chen, A. Ganapathi, R. H. Katz, "To Compress or Not To Compress Compute vs. IO Tradeoffs for MapReduce Energy Efficiency," Green Networking 2010.