Abstract—This paper aims at presenting the Big Data Integration and techniques available for Big Data Integration using Hadoop techniques. Data is collected and stored at unprecedented rates. Big data integration is the process of transferring data in source format into destination format. Many data warehouse and management are supported by integration techniques and transportation by using Extract Transform-Load process. Map Reducing method are used to extract and classify the data groups. Map takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs). Secondly, reduce task, which takes the output from a map as an input and combines those data tuples into a smaller set of tuples. Big Data Integration as a process is highly combined and iterative to add new data sources. The challenge is not only to store and manage the vast volume of data, but also to analyze and extract meaningful value from it. Big data comes from relatively new types of data sources like social media, public filings, and content available in the public domain through agencies or subscriptions, documents and e-mails including both structured and unstructured texts format.

Keywords—Map Reduce Techniques, Hadoop, Fault-Tolerance and High Availability

I. INTRODUCTION

Today, organizations invest more in data manipulation and most of the time the stored data are unused and they are not retrieved and utilized in a proper way. Enterprises and organizations are spending more in data processing. Actually big data is not a problem whereas it is a big asset to the organizations. THE term “Big Data” basically derived from the big or large volume of data[1]. It was defined as a situation where the volume, velocity and variety of data exceed organizations storage or compute capacity for accurate and timely decision making. Storage of these big data can be done by introducing multiple data centers, where as utilizing these data in an effective manner is a tedious one which is to be considered carefully. Data from different sources and integrating such data is very important. But in Big Data environment data from different sources are of different formats and existing data warehousing techniques are inefficient to handle such situation[2]-[3]. This paper aims at presenting the Big Data Integration and techniques available for integrating big data using Hadoop technique.

1.1 HADOOP

Doug Cutting and his team developed an Open Source Project called HADOOP is shown in fig.1. Hadoop is an Apache open source framework written in java that allows distributed processing of large datasets across clusters of computers using simple programming models. The Hadoop framework application works in an environment that provides distributed storage and computation across clusters of computers [4]. Hadoop is designed to scale up from single server to thousands of machines, each
offering local computation and storage. Hadoop runs applications using the MapReduce algorithm, where the data is processed in parallel with others. In short, Hadoop is used to develop applications that could perform complete statistical analysis on huge amounts of data.

![Hadoop Framework](image1)

**Figure 1:** Hadoop Framework

1.2 Hadoop Architecture
There are two major layers are present in the Hadoop architecture illustrate in the fig2. They are (a) Processing/Computation layer (MapReduce) (b) Storage layer (Hadoop Distributed File System).

![Hadoop Architecture](image2)

**Figure 2:** Hadoop Architecture

1.3 Advantages of HADOOP:
- Hadoop framework allows the user to quickly write and test distributed systems. It is efficient, and it automatic distributes the data and work across the machines and in turn, utilizes the underlying parallelism of the CPU cores.
• Hadoop does not rely on hardware to provide Fault-Tolerance and High Availability (FTHA), rather Hadoop library itself has been designed to detect and handle failures at the application layer.
• Servers can be added or removed from the cluster dynamically and Hadoop continues to operate without interruption.
• Another big advantage of Hadoop is that apart from being open source, it is compatible on all the platforms since it is Java based[5]-[7].

1.4 Big Data And Its Characteristics

Big data is defined as high-volume, high-velocity and high-variety information assets that demand cost-effective, innovative forms of information processing for enhanced insight and decision making. Big data handle five different types of V”s are - Volume, Velocity, and Variety, veracity and variability is exposed in fig.3.

Volume- Volume derives the amount of data from terabytes to Peta bytes.
Velocity- Velocity represents speed (ie.) Velocity means the rate of change in the data and how fast it must be processed to gain business value.
Variety - Big Data means much more than traditional RDBMS data. It includes Unstructured text, sound and movie files, images, documents, geo-location data, web logs, etc.
Veracity – Data in doubt or unpredictable data.
Variability - Semantics or the variability in language[8]-[12].

The Big Data are primarily categorized in to three types is displayed in fig.4. They are
• Structured data: Relational data.
• Semi Structured data: XML data.
• Unstructured data: Word, PDF, Text, Media Logs.
II. METHODOLOGY

The World Wide Web has rich source of voluminous and heterogeneous information which continuous to expand in size and complexity. Many web pages are unstructured and semi-structured. Information extraction is primary task in data integration process. Data Extraction takes data from source systems and makes it available to the data warehouse. Main objective of this extraction process is to extract right information from the right place at the right time with the right cost to support right decision. Traditionally, data warehouses were used as multidimensional databases to store large amounts of data. Poor data quality results in poor decision making which ultimately results in poor performance. Completeness, Consistency and Correctness of data supports organizations and is the cornerstone of decision making systems in organizations. Good functioning and efficiency of an organization depends on the quality of the data. Poor data quality decreases customers satisfaction and automatically degrades the performance and decision making ability [13]-[14]. Big data avoids the drawbacks from conventional methods by using this below mentioned method

2.1. Extract, Transform and Load Process

ETL comes from Data Warehousing and stands for Extract-Transform-Load is demonstrated in the fig.5

2.1.1 Extract

The “Extract” task involves gathering data from external sources that needs to be brought to the required systems and databases. The data might be found in stand-alone databases or spreadsheets that are not integrated with any master database.

2.1.2 Transform

In the “transform” step a variety of software tools and even custom programming are used to manipulate the data.

2.1.3 Load

After the successfully transformation of the source data it is required to physically load it into the target system or database. Before loading the data, it is required to make sure that there is a backup of the current system so that roll back or undo can be initiated in case of failure of the Load process. After loading the data, it’s common to run audit reports so that there can be review of the results of the merged databases and systems to make sure the new data hasn’t caused any errors.

Figure 5: The General Process of ETL
2.2 MapReduce Model

MapReduce is a processing technique and a program model for distributed computing based on java. The MapReduce algorithm contains two important tasks, namely Map and Reduce. Map takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs). Secondly, reduce task, which takes the output from a map as an input and combines those data tuples into a smaller set of tuples. As the sequence of the name MapReduce implies, the reduce task is always performed after the map job. The major advantage of MapReduce is that it is easy to scale data processing over multiple computing nodes is explained in fig.6.

![Figure 6. MapReduce Model](image)

Under the MapReduce model, the data processing primitives are called mappers and reducers. Decomposing a data processing application into mappers and reducers is sometimes nontrivial. But, once we write an application in the MapReduce form, scaling the application to run over hundreds, thousands, or even tens of thousands of machines in a cluster is merely a configuration change. This simple scalability is what has attracted many programmers to use the MapReduce model[15].

2.3 Algorithms

MapReduce program executes in three stages, namely map stage, shuffle stage, and reduce stage.

2.3.1 Map stage

The map or mapper’s job is to process the input data. Generally the input data is in the form of file or directory and is stored in the Hadoop file system (HDFS). The input file is passed to the mapper function line by line. The mapper processes the data and creates several small chunks of data.

2.3.2 Reduce stage

This stage is the combination of the Shuffle stage and the Reduce stage. The Reducer’s job is to process the data that comes from the mapper. After processing, it produces a new set of output, which will be stored in the HDFS [15]-[17].

2.4 Terminology

Some of the terminologies that are used to represent the Big Data techniques are

- **PayLoad** - Applications implement the Map and the Reduce functions, and form the core of the job.
- **Mapper** - Mapper maps the input key/value pairs to a set of intermediate key/value pair.
- **NamedNode** - Node that manages the Hadoop Distributed File System (HDFS).
- **DataNode** - Node where data is presented in advance before any processing takes place.
- **MasterNode** - Node where JobTracker runs and which accepts job requests from clients.
SlaveNode - Node where Map and Reduce program runs.
JobTracker - Schedules jobs and tracks the assign jobs to Task tracker.
Task Tracker - Tracks the task and reports status to JobTracker.
Job - A program is an execution of a Mapper and Reducer across a dataset.
Task - An execution of a Mapper or a Reducer on a slice of data.

III. RESULT AND DISCUSSION

Given below is the data regarding the electrical consumption of an organization from 2010 to 2014 is shown in table 1. It contains the monthly electrical consumption and the annual average for various years.

If the above data is given as input, we have to write applications to process it and produce results such as finding the year of maximum usage, year of minimum usage, and so on. This is a walkover for the programmers with finite number of records. So logic is written using MapReduce() technique with the help of java to produce the required output. If the above mentioned table is applied as input the application written in java changed the table in to sample.txt is shown in table 2.

OutPut:
INFO mapreduce.Job: Job job_1414748220717_0002
completed successfully
16/09/18 06:02:52
INFO mapreduce.Job: Counters: 49
File System Counters
FILE: Number of bytes read=61
FILE: Number of bytes written=279400
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=546
HDFS: Number of bytes written=40
HDFS: Number of read operations=9
HDFS: Number of large read operations=0
HDFS: Number of write operations=2
Job Counters
Launched map tasks=2
Launched reduce tasks=1
Data-local map tasks=2
Total time spent by all maps in occupied slots (ms)=146137
Total time spent by all reduces in occupied slots (ms)=441
Total time spent by all map tasks (ms)=14613
Total time spent by all reduce tasks (ms)=44120
Total vcore-seconds taken by all map tasks=146137
Total vcore-seconds taken by all reduce tasks=44120
Total megabyte-seconds taken by all reduce tasks=45178880
Map-Reduce Framework
Map input records=5
Map output records=5
Map output bytes=45
Map output materialized bytes=67
Input split bytes=208
Combine input records=5
Combine output records=5
Reduce input groups=5
Reduce shuffle bytes=6
Reduce input records=5
Reduce output records=5
Spilled Records=10
Shuffled Maps =2
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=948
CPU time spent (ms)=5160
Physical memory (bytes) snapshot=47749120
Virtual memory (bytes) snapshot=2899349504
Total committed heap usage (bytes)=277684224
File Output Format Counters
Bytes Written=40

<table>
<thead>
<tr>
<th>Years</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>July</th>
<th>Aug</th>
<th>Sep</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014</td>
<td>23</td>
<td>23</td>
<td>2</td>
<td>43</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>25</td>
<td>26</td>
<td>25</td>
</tr>
<tr>
<td>2013</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>30</td>
<td>31</td>
<td>31</td>
<td>31</td>
<td>30</td>
<td>30</td>
<td>29</td>
</tr>
<tr>
<td>2012</td>
<td>31</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>36</td>
<td>34</td>
<td>34</td>
<td>34</td>
</tr>
<tr>
<td>2011</td>
<td>39</td>
<td>38</td>
<td>39</td>
<td>39</td>
<td>39</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td>40</td>
<td>39</td>
<td>38</td>
<td>40</td>
</tr>
<tr>
<td>2010</td>
<td>38</td>
<td>39</td>
<td>39</td>
<td>39</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>00</td>
<td>40</td>
<td>39</td>
<td>39</td>
<td>45</td>
</tr>
</tbody>
</table>

Table 1. Electrical Consumption of an Organization from 2010 to 2014

Our proposed application using Java reads the table 2 and produces the results in the form of maximum usage year and minimum usage year with its month reading using MapReduce algorithm is depicted in the fig.7.
IV. CONCLUSION

Today the data created, generated, collected and analyzed a large scale creates problems during a management and manipulation. Big Data Integration is the major challenge in Big Data Environment due to its heterogeneity nature. This paper aims at giving the outline of challenges of data integration and the MapReduce techniques are used to integrate the big data to produce the good and accurate result. In future we are going to apply a new framework, with latest technology and proposed algorithm to improve the accuracy and timeliness of the given data.

REFERENCES


Author’s Biography

S. Ramkumar is currently working as an Assistant Professor at Kalasalingam University, KrishnanKoil. He received his MCA Degree in Karunya University and M.Phil. Degree in Karpagam University. He has five years of excellence in teaching and worked as Assistant professor in PG Department of Computer Science at Subramanya College of Arts and Science, Palani, and V.S.B Engineering College, Karur in Tamilnadu. He obtained his Doctorate degree in Computer science in Karpagam University, Coimbatore, Tamilnadu. His areas of interest include Data Structures, Operating System, Java, Web Programming, System Software, Object Oriented Analysis and Design, Software Engineering and Digital Signal Processing. He has published several papers in referred journals and conferences. His field of interest is Bio Signal Processing, Artificial Intelligence, Human Computer Interface, Brain Computer Interface and Machine Vision. He is an editorial board member of the several journals, in India. He has received gold, silver and bronze medals in National and International exhibitions for his research products on vision and Human Computer Interfaces.

G. Ganesaperumal is currently pursuing his MCA Degree in Kalasalingam University. He obtained his UG degree in Bachelor Computer Applications in Arulmigu Kalasalingam College of Arts & Sciences, Krishnankoil, Tamilnadu. His areas of interest include Data Structures, Java, and Software Engineering.