BIG DATA ANALYTICS WITH LONG RANGE PLAN TO PROCESS LARGE DATA SETS

N.PADMAJA[1], Prof. T.SUDHA[2]

[1]Department of Computer Science and Engineering,
School of Engineering & Technology,SPMVV,Tirupati
[2]Depatment of Computer Science,SPMVV,Tirupati

[1]gowripadma@yahoo.com
[2]thatimakula_sudha@yahoo.com

Abstract---- BIG DATA is extremely large data sets may be analyzed computationally to reveal patterns, trends and associations especially relating to human behavior and interaction the traditional way of processing huge datasets has been shifted from centralized architecture to distributed architecture A generic application and robust analytic system is developed for collecting and delivering high volumes of data Big data incorporates ideas from existing centralized system and shifted to distributed system. Using hadoop framework which stores large amount of unstructured data. To design a scalable approach to this system, Map Reduce Distributed processing framework it is a two phase process that is transformation and aggregation in which the application is divided into many small fragments of work, each of which may be executed or re-executed on any node in the group, then the analyzed data can be send without any data loss. Our expected results will produce superior performance depend on criteria of application. Result will be displayed in any format as per organizations requirements.

Index Terms---Hadoop Map Reduce, scalable, aggregation, transformation

I. INTRODUCTION

Big Data is large volumes of data sets that may be analyzed computationally to reveal patterns, trends and associations, especially relating to human behavior and interactions. Big Data Analytics is a process of examine large data sets containing variety of data types i.e. Big Data to uncover hidden patterns, unknown correlations, market trends, customer preferences and other useful information. The proposed application for big data analytic, where the traditional way of processing huge datasets has been shifted from centralized architecture to distributed architecture. Big data is a generic application, multiple data cluster data nodes are produces one output after implementing Map Reduce job, finally the best result is provided by name node which monitor whole flow. Hadoop provide scalable approach to this application. Map Reduce Distributed processing framework where the application is divided into many small fragments of work, each of which may be executed or re-executed on any node in the cluster. Hadoop is a merging domains that is distributed system and big data. Hadoop has 100% guarantee to secure of data and scope.

Hadoop system acts to distribute data across a network or drift the data as necessary. Hadoop is a open source program under Apache license that is maintained by global community of users.

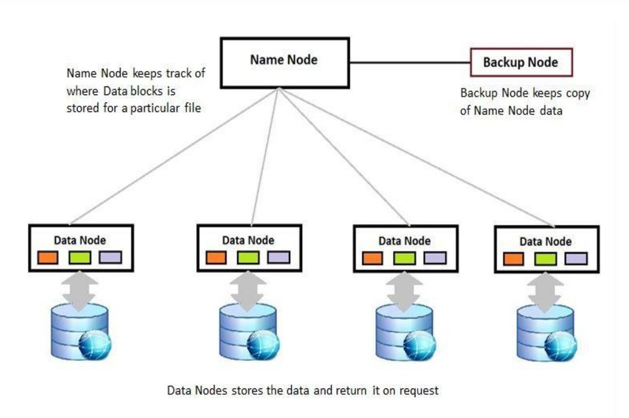
Name Node
Name Node keeps track of where Data blocks is stored for a particular file
Backup Node
Backup Node keeps copy of Name Node data
Data Node
Data Node
Data Node
Data Node
Data Nodes stores the data and return it on request

Figure 1.Data Nodes stores the data and return on request

- **Name Node:**
  It is the admin / master of the system.
  - It manages the blocks which are present on the Data Node.
  - Name Node stores the meta-data and it runs on high quality hardware.
  - It is single entry point for any of the failures happening in Hadoop Cluster.

- **Data Node:**
  These are the slaves which are deployed on each machine.
  - This is the place where actual data is stored.
  - It is` responsible for serving read and write requests for the clients.
  - Each Data Node holds a part of the overall data and processes the part of the data which it holds.

Backup Node:

Backup Node is responsible for performing periodic checkpoints. When the Name Node failure occurs, it can be restarted with the Name Node using the checkpoint.

A. HADOOP DISTRIBUTED FILE SYSTEM (HDFS)

Hadoop is one of the tools design to handle Big Data. There are several components within the Hadoop environment performing data management operations. HDFS is based on Google file system and provides a distributed file system that is to design on commodity hardware and low cost hardware and fault tolerance. Hadoop yield high through put axis to application data and is suitable for applications having extremely large data sets.

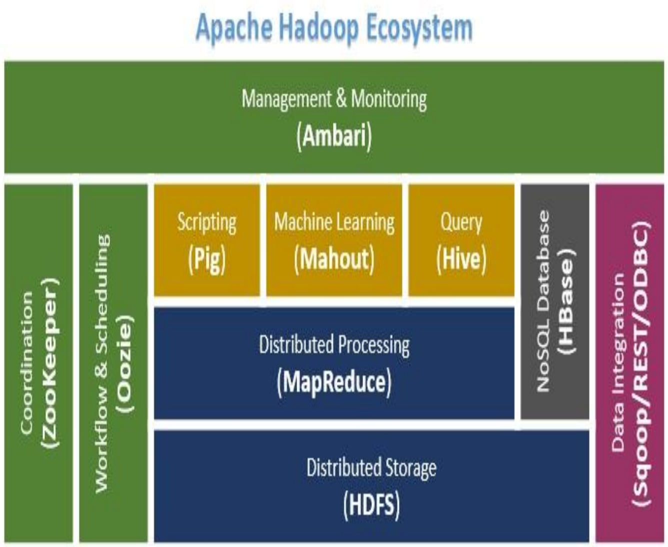

Figure-2 : HDFS architecture

B. Zookeeper

Zookeeper is a centralized service to prolong configuration information, naming, presuming distributed synchronization, and understanding group services. These kinds of services are used in distributed applications. Each time it is implemented there is a lot of work that goes into focusing the bugs and race conditions that are unavoidable. Because of the difficulty in implementing these kinds of services, applications initially brief on them, which make brittle in the existence of change it is difficult to manage. When it is done correctly, other implementations of these services lead to management complexity for the deployment of applications. Zookeeper is a thread and also demon process. Data Node: actual work is done by the data node. Name Node: If name node will dead then it will give task to one of the data node which is nearer to that name node.

C Oozie

Apache Oozie is a server-based workflow scheduling system to manage Hadoop jobs Workflows in Oozie are defined as a collection of control flow and action nodes in a directed acyclic graph(DAG). Control flow nodes determine the beginning and the end of a workflow (start, end and failure nodes) and implement to control the workflow execution path (decision, fork and join nodes). Action nodes process a workflow execution of a computation/processing task. Oozie provides support for different types of actions including Hadoop Map Reduce, Hadoop distributed file system operations, Pig, SSH, and email. Oozie can also be incorporated to support additional action types for the execution.

.

D Pig

Apache Pig is Latin language, high-level platform for creating software programs that run on Apache Hadoop. The language for this platform is called Pig Latin. Pig can execute its Hadoop jobs in Map Reduce, Apache Spark. Pig Latin extracts the programming from the Java Map Reduce jargon which makes Map Reduce programming high level, similar to that of SQL for RDBMSs. Pig Latin can be incorporated using User Defined Functions (UDFs) in which the user can write in Java, Python, JavaScript, Ruby or Groovy and user can call user defined functions directly from the Pig language it self..

E Mahout

Apache Mahout is a project of the Apache Software Foundation to produce free implementations of distributed or otherwise scalable machine learning algorithms focused primarily in the areas of collaborative filtering in which user can build personalized recommendations on the web , clustering and classification. Most of the implementations use the Apache Hadoop platform. Mahout also provides Java libraries for common math's operations (focused on linear algebra and statistics functions) and primitive Java collections. Mahout is a work in progress; the number of implemented algorithms has grown quickly for the computations.

F Hive

Apache Hive is a data warehouse infrastructure built on upper most layer of Hadoop for providing data summarization, query, and analysis. Hive gives an SQL-like interface to query data stored in various databases and file systems that combine with Hadoop. Traditional SQL queries must be implemented in the Map Reduce Java API to execute SQL applications and queries over distributed data. Hive provides the necessary SQL abstraction to merge SQL-like Queries (Hive QL) into the underlying Java API without the need to implement queries in the low-level Java API

G H base

H Base is an open source, non-relational, distributed database modeled after Google's BigTable and is written in Java. It is developed as part of Apache Software Foundation's Apache Hadoop project and runs on top of HDFS (Hadoop Distributed File System), providing Big Table-like capabilities for Hadoop. H base is a fault-tolerant way of storing large quantities of sparse data (small amounts of information caught within a large collection of empty or unimportant data, such as extracting the 50 largest items in a group of 2 billion records, or finding the non-zero items representing less than 0.1% of a huge collection).

H Map Reduce

A **Map Reduce** job usually splits the input data-set into independent small pieces of data which are processed by the map tasks in a completely parallel manner. Map Reduce framework sorts the outputs of the maps, which are then input to the reduce tasks. The major advantage of Map Reduce is easy to scale data processing over numerous computing nodes. In Map Reduce 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 Map Reduce form, scaling the application to run over hundreds, thousands, or even tens of thousands of machines in a cluster is merely a configuration change. The scalability of Map Reduce has attracted many programmers to use the Map Reduce model.

The Algorithm

- Generally Map Reduce pattern is based on sending the computer in which the data resides

- Map Reduce program executes in three stages, namely map stage, shuffle stage, and reduce stage.

○ **Map stage** : The map or mapper's job is to process the input data. The input data is in the form of file or directory and is stored in the Hadoop file system (HDFS),file is passed to the mapper function one line after one line. The mapper processes the data and creates several small pieces of data.

○ **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.

- During a Map Reduce job, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster.
- The framework manages all the details of data-passing such as issuing tasks, verifying task completion, and copying data around the cluster between the nodes.
- Most of the computing takes place on nodes with data on local disks that reduces the network traffic.

- After completion of the given tasks, the cluster collects and reduces the data to form an appropriate result, and sends it back to the Hadoop server.

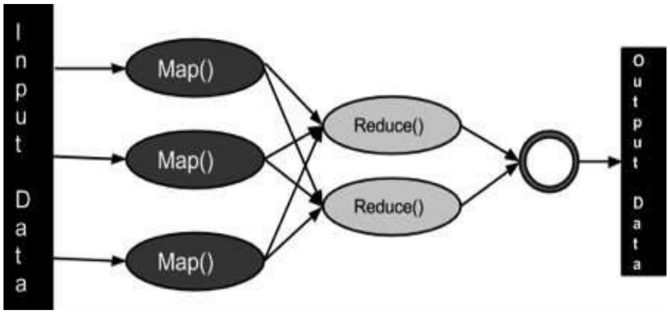

Figure:3 Map Reduce Phases

I HDFS

Apache Hadoop consists of a storage part, known as Hadoop Distributed File System (HDFS), and a processing part which is a Map Reduce programming model. Hadoop splits files into large blocks and distributes them across nodes in a cluster.

J Data Integration

Data Integration is a key step in a Hadoop solution architecture. Horton works with Talend to bring an open source integration tool for easily connecting Apache Hadoop to hundreds of data systems without having to write code. Talend Open Studio for Big Data is a powerful and versatile open source solution for big data integration that natively supports

Apache Hadoop, including connectors for Hadoop Distributed File System (HDFS), H Base, Pig, Sqoop and Hive.

Sqoop is a tool designed to transfer data between Hadoop and relational databases. Sqoop is to import data from a relational database management system (RDBMS) such as My SQL or Oracle into the Hadoop Distributed File System (HDFS), transform the data in Hadoop MapReduce, and then export the data back into an RDBMS. Sqoop automates most of this process depending on the database to report the schema for the data to be imported. Sqoop uses Map Reduce to import and export the data, which provides parallel operation as well as fault tolerance.

II PARSER

Parser, which Parse multi-structured and industry-standard data on Hadoop, including industry standards documents, log files, and complex file formats. Multiple heterogeneous file format like xml file, text file, xls file, etc. are converted into standard CSV (comma-separated values).file format. CSV file extension has become a kind of legal industry standard. The information is organized with one record on each line and each field is separated by comma.CSV file is a set of database rows and columns stored in a text file such that the rows are separated by a new line while the columns are separated by a semi colon or a comma.

The advantage of using CSV file format for data exchange is that the CSV file is relatively easy to process by any application and data extraction can be achieved with the help of a simple program. When database technologies where in developing stage , the CSV was the most standard portable format. A CSV file is a way to collect the data from any table can be conveyed as input to another table-oriented application such as a relational data base application. Microsoft Excel, a leading spread sheet or relational database application, can read CSV files. A CSV file is also called as a flat file.

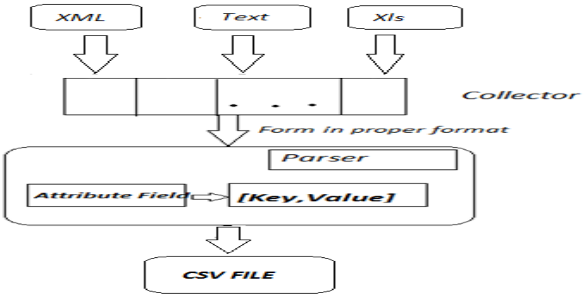

Figure:4 CSV file

III CUSTOM PARTITIONING

The partitioning phase occurs after the map phase and before the reduce phase. The number of partitions is equal to the number of reducers. The data gets partitioned across the reducers according to the partitioning function.The difference between a partitioner and a combiner is that the partitioner divides the data according to the number of reducers, the data in a single partition gets executed by a single reducer. The output from map is then feed to reduce tasks which processes the user defined reduce function on map outputs, before the reduce phase another process that partition the map outputs based on the key value and keeps the record of same key into the partitions by default the partitioner implements Hash Partitioner. It uses the hash Code() method of the key objects modulo the number of partitions total to determine which partition to send a given (key, value) pair to. Partitioner provides the getPartition() method can implement the custom partition for a job.

The getPartition() method receives a key and a value and the number of partitions to split the data, a number in the range [0, numPartitions) must be returned by this method, indicating which partition to send the key and value to.

For any two keys k1 and k2, k1.equals(k2)
implies getPartition(k1, *, n) == getPartition(k2, *, n).

IV CONCLUSION

Big data incorporates ideas from existing centralized system and shifted to distributed system. Using hadoop framework which stores large amount of unstructured data. Big Data Analytics is a process of examine extremely large data sets containing variety of data types i.e. Big Data to uncover hidden patterns, unknown correlations, market trends, customer preferences and other useful information. HDFS is based on Google file system and provides a distributed file system that is to design on commodity hardware. It has to be low cost hardware and fault tolerant. The data gets partitioned across the reducers according to the partitioning function.The difference between a partitioner and a combiner is that the partitioner divides the data according to the number of reducers so that all the data in a single partition gets executed by a single reducer. The output from map is then feed to reduce tasks which processes the user defined reduce function on map outputs.

References

[1] "Big Data: A Revolution That Will Transform How We Live, Work, and Think"**by Viktor Mayer-Schönberger**, **Kenneth Cukier** Published March 5th 2013 by Houghton Mifflin Harcourt

[2] "Big Data Imperatives " Enter prise Big Data Warehouse,BI Implementations and Analytics by Soumendra Mohanty, Madhu Jagadeesh,Harsha Srivatsa Apress

[3] **http://www.webopedia.com/TERM/B/big_data.html**

[4] Hadoop: The Definitive Guide, 4th Edition Storage and Analysis at Internet Scale By Tom WhitePublisher: O'Reilly Media March 2015

[5] Mahran, Ahmed (30 October 2015). "Oozie Plugin for Eclipse". www.infoq.com. InfoQ. October 2015.