 ISBN:978-93-5254-640-4

Big Data Analysis inside A DBMS Tightly Coupled With Different File Systems Perspectives.

Anurag Kataria

Research Scholar, RTU, Kota

Abstract ***DBMS is considered important analytics for Big Data having high functionality of query languages, indexes or different schemas to maximize scalability and parallelism. RDBMS remain the first order data management technology while many non-DBMS tools like statistical languages, generalized data mining techniques and large scale parallel systems are serving as the main technology for Big Data analytics. There has been a lot of research on DBMS support in MapReduce. The only technology which directly exploits the DBMS for Big Data analytics in the MapReduce framework is HadoopDB. It is taking advantages of both these technologies of DBMS and MapReduce there is a limitation that sharability is not supported by HadoopDB for entire data as it uses multiple nodes to save the data in a shared –nothing manner. Large scale systems have to use the base of HadoopDB and MapReduce hence DBMS seems not to be good technology to analyze Big Data despite of a fast and reliable data repository and handling SQL queries. HadoopDB cannot process queries efficiently that needs inter and intra node communications. It means HadoopDB must have to reload the whole data to handle some queries or cannot handle some complex queries. In my research effort I propose a NFS-integrated DBMS where a DBMS is tightly coupled with networked file system (NFS) through which we can achieve the sharability of the entire data. In my search I'm explaining the networked analytics on large Databases inside a DBMS. Although DBMSs cannot replace the parallel systems like MapReduce for web-scale textual data analysis. Here the technologies are influencing themselves each other. To process big data analytics parallel, I implement MapReduce framework on top of NFS-integrated DBMS. I also propose the notion of networked mapping for optimization. It will show that limitations of HadoopDB are overcomed by these strengths – (1) it will perform faster query processing as it will not needed to reload the data. (2) it will support more complex query types. Here I want to conclude with a proposal of research issues at long-terms taking into consideration of "Big Data Analysis" research work trends.***

Keywords ***Big Data Analysis, MapReduce, DBMS, Networked File System, HadoopDB.***

I. INTRODUCTION

Google data processing and storage is implemented by Hadoop which is a open source technology. A programming model is used by Hadoop called Map and Reduce that was used in the functional programing languages like LISP but if require, all data could not be loaded into memory. Big data term describe the exponential expansion and accessibility of structured and unstructured data. Traditional databases and other analytical techniques can not process the Big Data as it have larger occupation. Better decisions can only be taken when data is available in extremely large sizes for more and more accurate analyses. It creates a confident decision making which generates operational efficiencies, cost feasibility and better risk management.

Researchers and academicians are challenged to analyzing big data that needs special analytical technical skills. Big data analysis uncovers the patterns, correlations and other knowledge for betterment of the decision making. It gives us a recognition which data is most useful for future needs.

Hadoop Map Reduce technique analyses big data. Due to its high scalability it emerged as a latest paradigm for large scale data analysis, fine grain fault tolerant having easy algorithm implementation. MapReduce refers two tasks map and its reduction.

HDFS, the Hadoop Distributed File System runs on commodity hardware. HDFS is defined for bigger data (TB, PB or ZB) and provide high throughput accessing to the knowledge and information sharing.

II. OBJECTIVES AND CHALLENGES

The objective of multi-dimensional data analysis is to efficient prediction of future observations and to have the capability to get the relationship of outcomes with provided inputs for scientific objectives. Due to large samples big data have two additional goals of heterogeneity and generosity through distinct subpopulations. Big data satisfy two basic requirements of uncover the structures of each subpopulation of data while sample size is comparatively small that is not feasible traditionally. And the other one is features extraction commonly spread across subpopulations.

speed with accuracy is a challenge to make decisions fast but the enormous data volume and accessing the desired level of metadata is a cumbersome job as a challenge. This challenge increases when granularity degree increases. To explore big voluminous data in real time we can use enhanced memory or parallel processing or we can insert the data into memory. Clustered data can resolve this demand of speed and accuracy by making the data groups smaller to visualize the data effectively.

if the data is inaccurate and time lapsed, the value of that data is suspected and compromising for decision making purposes even though the accessing is speedy. This challenge can be overcome when big data analysis is used with some profound data visualization techniques. The graphical data can be generated using visualization techniques which can set trends and outliers fasters than relational tables having numeric and textual data. The analysts can easily spot the attention seeing simply at a chart representation prepared by some graphical

drawing techniques and methods.

it requires a lot of diversification to get the data in desired shape so that we can visualize that data for analysis purposes. Different geographical boundaries generate different nature of data having same meaning and understanding. We need to know the context, location, time stamp to analyze properly. What was the source of data and who is the consumer play the major role to mold the analysis process accordingly to interpret the information.

meaningful outcomes is another challenge when we have to deal with plotting of enormous volume of data on a graph for analyzing comparison the outcome results with data points on graph.

III. APPLICATIONS

Big data is used in so many diversified application areas like social networking platforms data analysis. A huge amount of data is generated by these platforms of LinkedIn, YouTube, Twitter, Facebook, etc. the data generated by these platforms can be exploited by several means using so many individual's characteristics. It may contain user's preferences. The confidential data leads economic indicators, business intelligence and societal socio-political states.

Customized personal services can be generated and implemented by commercial service oriented sector if they have the prediction about buyer's requirements by collecting their transaction-records.

Network-security is important where identifying the source of attack is high priority objective. Historical data and artificial intelligence are efficient enough to explore and disclose the attacker's location, target and probable loss.

Some health parameters can be revealed by big data analysis such as a person's body characteristics, his activities, environmental factors, etc. to diagnose a disease and selection of treatment accordingly that will be customized for individual.

Many archives are digitized now. Archives have billions of scanned books and earmarked almost every word of these books. It produces the bulk data volumes in chunks to be studied for a specific topic or analyzing historical events like study on coral-reefs in indo-shrilankan Ocean made during lord shri ram age.

IV. HADOOP FILE SYSTEM

Hadoop Distributed File System became very popular after social platforms like Twitter and Facebook used it to analyze interactive data.

i. Why Hadoop?

Hadoop provides Scalable, Economical, easy Computable and Storage-able platform. Hadoop brings a new way to store and analyze data. Since it is linear scalable on low cost commodity hardware, it removes the limitation of storage and compute from the data analytics equation. Instead of pre-optimizing data in the traditional ETL, data warehouse, and BI architecture, Hadoop stores all of the raw data and applies all transformation and analytics that it might be done on demand. Think of a traditional, static schema database as cache, that thanks to Hadoop, we don't need anymore.

ii. Hadoop: optimized for analytics, Comparing random vs. sequential

Apache Hadoop is optimized for analytical workloads. The MapReduce programming model is designed for analytics and the Hadoop file system is optimized for sequential data access. On the other hand, traditional RDMS databases are purpose built and optimized for record storage and retrieval with random read and write access. Thus, Hadoop is magnitudes faster for analytics workloads that need to scan through all the data like joins and aggregations.

iii. Small and Big Data

Hadoop's low-level optimization for analytic workloads makes it a powerful platform on individual computers as well as clusters of machines. Optimized sequential data access is not only faster on normal hard drives but also on new Solid State Drives and even outperforms in-memory random data access. Therefore, using Hadoop for small datasets on your desktop makes sense since your data may grow or you may develop data analytics that one day will run against a larger data set on a cluster.

iv. Choose the best Hadoop for you : Hadoop Distributions

Hadoop recently became very popular with several different vendors like Apache, Amazon, Cloudera, EMC, Hortonworks, IBM, MapR, Microsoft, etc. offering distributions with a set of optimizations and features. These vendors are committed to supporting all of the Hadoop distributions and allow easy migration from one to other. These isolates the end user from the lower level technical details and provides an simple though powerful web bases application on top that abstracts all interactions with Hadoop.

v. New Architecture

Hadoop files are saved redundantly on multiple nodes for durability on failure and for availability to every parallel process. HDFS implements master/slave architecture. Hadoop cluster consists of a NameNode, which is a master server to manage the file system namespace and it regulates accessing the files. There are a number of DataNodes which manage storage attached with nodes. Physically, a file is divided in one or more blocks and these are managed in a set of DataNode. The NameNode performs operations like opening, closing, and renaming files and directories. DataNodes serves read and write requests. DataNodes can creates blocks, deletion, and replication when instructed by NameNode. The Namenode actively monitors the number of replicas of a block. When a replica of a block is lost due to a DataNode failure or disk failure, the NameNode creates another replica of the block. Hadoop use the Java language to built the file system. any machine supporting Java can run the NameNode or the DataNode software.

V. MAPREDUCE

MapReduce is easily parallelizable, scales linearly and is highly optimized for analytical workloads. MapReduce is a framework for the analysis of big data on a large number of servers. It was originally designed for the back end of Google's search engine to enable a large number of commodity servers for efficient analysis of webpages collected in a large quatity.

It works on the principle of parallel and distributed processing. Earlier it was extensively used with its open source technique Hadoop. MapReduce implements master/slave model in its implementation. MapReduce divides a process into sub-

processes which executed in parallel fashion and the results are then aggregated to get the final outcome. Programmers need not to know the implementation details of parallelism, mapreduce automatically maintain parallel processing for the programs written in mapreduce.

There are two algorithms of two different functions: map and reduce. The map function performs the read operations and supplies the data to the reducers. The Read and Write operation of map and reduce functions worked as key-value pairs.

MapReduce algorithm implements the Driver which initializes the process with its configuration information and assigns the specific mapper-reducer classes. It informs the platform to run the code on the input file(s) and directing the pointer for the output files.

When large data sets are being processed, for each input of logical record it use a mapping function Mapper to generate key value pairs. The reducer is applied to the shared data which have the same key.

i. Mapper The mapper generates an random number for key-value pairs by applying this function on each input key-value pair. The mathematical representation of this implementation is as under:

Map (Key, Value) →→→ List (intermediateKey, intermediateValue)

The main objective of the mapper is to manage the data for the processing in the reduce phase. The input of mapper is key-value pairs. By default, the value is a data record and key is offset of the logical data record. The resultant outcome is a bunch of key-value pairs which works now as a input for the reducer function. To optimize the processing capacity, MapReduce runs multiple identical executable mappers in parallel.

ii. Reducer The reducer is implemented on all values related with the same intermediate key for the generation of output key-value pairs.

Reduce (intermediateKey, list(intermediateValue)) → → → list (outKey, outValue)

Every reducer processes the intermediate values for a specific key produced by the mapper. there must be a one-to-one mapping between keys and reducers. Multiple reducers run in parallel. How many numbers of reducers will be run in parallel is decided by the user. By default, it is one reducer only.

VI. CONCLUSIONS

In concluding step we can say although data processing task have to be done on large growing data volumes but the era of big data analysis tools like MapReduce, Haddop, HDFS ensures us for faster advances in several science and its disciplines to improve the feasibility, profitability and entrepreneurial success.

These technologies have penetrations in data mining, information retrieval, image retrieval, machine learning, and pattern recognition. However growing data needs may limits their uses and suitability.

This paper explore mapreduce for big data analysis efficiently and resolving the processing complexities involved with large datasets in diversified domains. Mapreduce handles the scaling of an application effortlessly from a single machine to several parallel machines while fault tolerance and high performance issues are dealt with very good and enough efficiency.

VII. REFERENCES:

[1]. Hadoop,"PoweredbyHadoop". http://wiki.apache.org/hadoop/PoweredBy

[2]. Apache: Apache Hadoop, http://hadoop.apache.org

[3]. Jianqing Fan1, Fang Han and Han Liu, Challenges of Big Data analysis, National Science Review Advance Access published February, 2014.

[4]. Amazon simple storage service(Amazon S3). http://aws.amazon.com/s3/

[5]. Hadoop MapReduce. http://hadooptutorial.wikispaces.com/MapReduce

[6]. GrzegorzMalewicz, Matthew H. Austern, Aart J. C. Bik, James C.Dehnert, Ilan Horn, NatyLeiser, and GrzegorzCzajkowski,Pregel: A System for Large-Scale Graph Processing, SIGMOD10, June 6– 11, 2010, pp 135-145.

[7]. Guoping Wang and CheeYong Chan, MultiQuery Optimization in MapReduce Framework.

[8]. Jeffrey Dean and Sanjay Ghemawat, MapReduce: Simplified Data Processing on Large Clusters.

[9]. Changqing Ji, Yu Li, Wenming Qiu, Uchechukwu Awada, Keqiu Li, Big Data Processing in Cloud Computing Environments, 2012 International Symposium on Pervasive Systems, Algorithms and Networks.

[10]. Jens Dittrich JorgeArnulfo Quian´eRuiz, Efficient Big Data Processing in Hadoop MapReduce.

[11]. Apache Hive, http://hive.apache.org/

[12]. HadoopTutorial YahooInc. https://developer.yahoo.com/hadoop/tutorial/index.html

[13]. Kyuseok Shim, MapReduce Algorithms for Big Data Analysis.

[14]. Apache Giraph Project, http://giraph.apache.org/

[15]. OnurSavas, YalinSagduyu, Julia Deng, and Jason Li,Tactical Big Data Analytics: Challenges, Use Cases and Solutions, Big Data Analytics Workshop in conjunction with ACM Sigmetrics 2013,June 21, 2013.

[16]. Hadoop Distributed File System (HDFS). http://hortonworks.com/hadoop/hdfs/

[17]. VinayakBorkar, Michael J. Carey, Chen Li, Inside "Big Data Management":Ogres, Onions, or Parfaits?, EBDT/ICDT 2012 Joint Conference Berlin, Germany, 2012 ACM 2012, PP 3-14.

[18]. Dr.Siddaraju, Sowmya CL, Rashmi K, Rahul M "Efficient Analysis of Big Data Using Map Reduce Framework" International Journal of Recent Development in Engineering and Technology www.ijrdet.com (ISSN 2347-6435(Online) Vol2, Issue6, June2014)