Review Article | Impact Factor: 0.621 | ISSN: 2319-507X
Pramod Patil, IJPRET, 2014; Volume 2 (8): 201-207 | IJPRET

INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

A REVIEW ON VIRTUAL DATABASE SYSTEM USING MAP-REDUCE TECHNOLOGY

MR. PRAMOD C. PATIL, MR.VIJAY B. PATIL

Department of Computer Engineering, Marathwada Institute of Technology, Aurangabad (MS) India.

Accepted Date: 15/02/2014 ; Published Date: 01/04/2014

Abstract: Data Integration in the cloud and grid computing is playing very important role in many applications and research. Many algorithms and systems are designed and developed to address these issues. Virtual database systems are one of the effective solutions for data integration. The existing solutions to design virtual database systems are not so effective. Map Reduce is a computing model specifically designed for processing huge datasets on distributed sources with parallel processing and also has a good performance on large-scale data processing. In this paper, we propose a new distributed data integration system, as VDB-MR. Currently Hadoop is successfully applied for file based datasets. Now we want to utilize the parallel and distributed processing capability of Hadoop Map Reduce to handle heterogeneous query execution on large datasets to efficiently integrate heterogeneous data.

Keywords: Database integration; Virtual Database technology; Hadoop Map Reduce; Heterogeneous Databases; VDB-MR

Corresponding Author: Mr. PRAMOD C. PATIL

Access Online On:

www.ijpret.com

How to Cite This Article:

Pramod Patil, IJPRET, 2014; Volume 2 (8): 201-207

PAPER-QR CODE

INTRODUCTION

Now a day's the Data is stored in geographically decentralize manner. It may have heterogeneous data sources. Data integration is used to combine data from different resources and provide users with a unified view of these data. Virtual database[1] is one of the solution for data integration. Various techniques are used to design a virtual database. But most of them concentrates on how a global schema is extracted from autonomous resources, how an effective query language is identified, and how decomposing queries are optimized. Query execution has received little attention.

The Map Reduce software framework[4-6] has well-proven success in Google and Hadoop[7] communities. Currently Hadoop is applicable for file based datasets. This paper proposes how to utilize the processing capability of Hadoop Map Reduce to handle heterogeneous query execution for large datasets.

Virtual Database System

The data may be stored in several decentralize locations. Virtual Databases differ from standard databases because data are not really stored into the database. Virtual databases gives us a way to query and integrate this data. We present here a VDB system which focuses in the reuse of the information available on the Web, providing programmers with an easy and quick way to use that information

A virtual database system is a type of meta-database management system[2] which transparently maps multiple autonomous database systems into a single virtual database. The data sources are interconnected through computer network and may not be geographically centralized. Since the basic database systems remain autonomous, a virtual database system is an alternative to the task of merging several dissimilar databases.

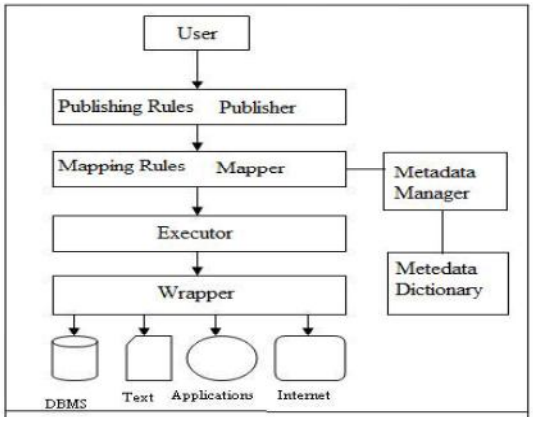

Fig.1 Virtual Database system

To this end, a virtual database system must be able to decompose the query into subqueries for the relevant essential Data, after which the system must composite the result sets of the subqueries. Because various database management systems employ different query languages, virtual database systems can apply wrappers [2] to the subqueries to translate them into the suitable query languages.

The main difference between a virtual database and a conventional database is that conventional database contains data, whereas a virtual database points to other databases that contain the data [2]

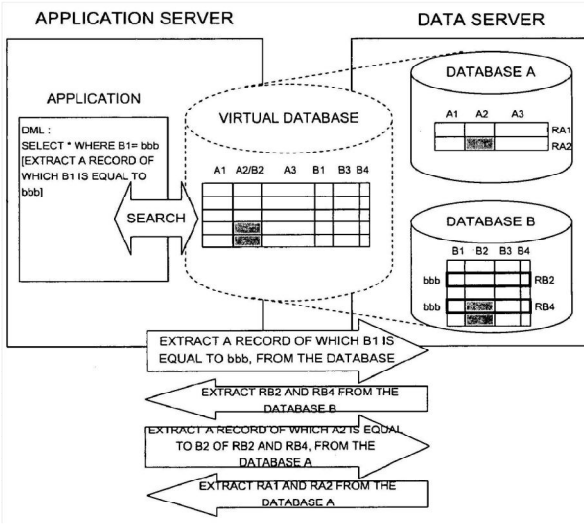

Fig.2 Extracting data using virtual database

Map Reduce Technique

MapReduce[3-7] is a software framework to process large data sets in a distributed fashion over a several machines. The core idea behind MapReduce is mapping your data set into a collection of <key, value> pairs [3-7], and then reducing overall pairs with the same key. The overall concept is simple, but is actually quite expressive

A MapReduce job usually splits the input data-set into independent chunks which are processed by the map tasks in a parallel manner. The framework sorts the outputs of the maps, which are then applied as input to the reduce tasks. Typically both the input and the output of the job are stored in a file-system. The master[6] takes care of scheduling tasks, monitoring them and re-executes the failed tasks.

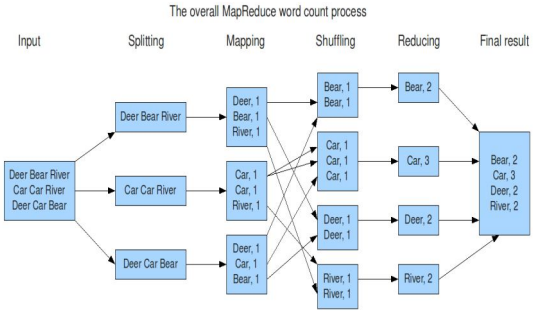

Fig 3. MapReduce execution Process

The MapReduce framework consists of a single master JobTracker and one slave TaskTracker per cluster-node. The master is responsible for scheduling the jobs' component tasks on the slaves, monitoring them and re-executing the failed tasks. The slaves execute the tasks as directed by the master.

Apache's Hadoop and HDFS

Apache Hadoop is an open source software framework that supports data-intensive distributed applications. It enables applications to work with thousands of computational independent computers and petabytes of data. Hadoop was derived from Google's MapReduce and Google File System (GFS) technology[5]. Hadoop is a top-level Apache[4] project being built and used by a global community of contributors, written in the Java programming language. Yahoo! has been the largest contributor to the project, and uses Hadoop extensively across its businesses.
A MapReduce job is a unit of work that the client wants to be performed: it consists of the input data, the MapReduce program, and configuration information. Hadoop runs the job by dividing it into tasks, of which there are two types: map tasks and reduce tasks. There are two types of nodes that control the job execution process: a jobtracker and a number of tasktrackers. The jobtracker coordinates all the jobs run on the system by scheduling tasks to run on tasktrackers. Tasktrackers run tasks and send progress reports to the jobtracker, which keeps a record of the overall progress of each job. If a tasks fails, the jobtracker can reschedule it on a different tasktracker. Hadoop divides the input to a MapReduce job into fixed-size pieces called input splits, or just splits. Hadoop creates one map task for each split, which runs the user defined map function for each record in the split.
VDB-MR (Praposed System Design)

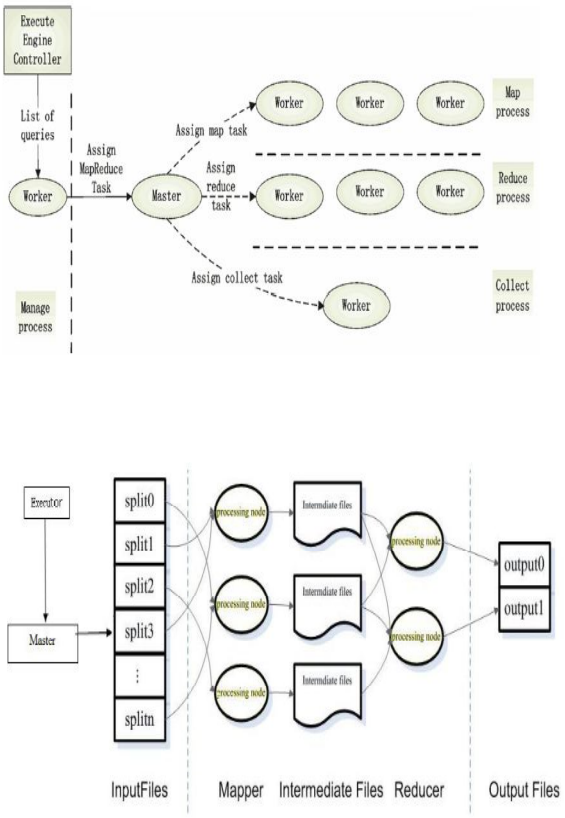

Fig 4.Structure of VDB-MR execution engine

The overall architecture of the distributed data integration system we propose in this paper, VDB-MR, is presented in Fig. which is based on the virtual database system. The architecture of VDB-MR is similar to the typical structure of virtual database presented in Fig. 4. The objective of VDB-MR is to integrate data in disparate databases and file systems and provide a uniform access to users. As the brain of the system, the parser engine manages the global schema which specifies the way users perform mapping and queries. The execute engine is however the body of the system. It collects metadata from member resources, optimizes and executes query processing.

The goal of distributed query processing is to execute such queries as efficiently as possible in order to minimize the response time that users must wait for answers or the time application programs are delayed. And to minimize the total communication costs associated with a query, to improved throughput via parallel processing, sharing of data and equipment, and modular expansion of data management capacity. In addition, when redundant data is maintained, one also achieves increased data reliability and improved response time.

This can be achieved through introducing MapReduce with VDB. In order to improve the performance efficiency of the VDB the Hadoop MapReduce is added at the executor phase. The executor will pass the Mapper's sub query to the Master of the MapReduce. The master will automatically split the input into chunks (splits) and finds M Mapper's and R reducers. The splits can be processed in parallel by the Mapper's. Reduce invocations are distributed by partitioning the intermediate key space into R pieces using a partitioning function. The number of partitions (R) and partitioning functions are specified by the user. The output of the R Reducers stored in R output files. This output files will fit our needs.

Advantages of Using Hadoop

- To overcome the shortcomings of RDBMS we can use Hadoop mapreduce technique
- RDBMS stores data in a set of tables with rows and columns, which means that an RDBMS enforces a strict structure when loading data.
- In RDBMS depending on the complexity of the process and the volume of data response time decreases.
- Hadoop is used to handle large amounts of work across a set of machines.
- Hadoop enables applications to work with thousands of nodes and petabytes of data.
- Hadoop is a open source implementation.
- MapReduce is a new framework specifically designed for processing huge datasets on distributed sources.
- Map and Reduce techniques to break down the parsing and execution stages for parallel and distributed processing.

Conclusion

Data integration is used to combine data from different resources and provide users with a unified view of these data. Virtual database is one of the solution for data integration. Various techniques are used to design a virtual database. But most of them concentrates on how a global schema is extracted from autonomous resources, how an effective query language is identified, and how decomposing queries are optimized.

Query execution has received little attention. Also the MapReduce software framework has well-proven success in Google and Hadoop communities. It is also successfully applied in the database field, such. However, these works mainly focus on isomorphic data resources stored in the form of key-value. To the best of our knowledge, no work has been done to apply MapReduce for data integration of heterogeneous data resources such as database or file systems.

So in this paper, we proposed a new data integration approach of a virtual database by utilizing the MapReduce technology which translates a query into sub queries by the global schema, executes them and merges the results concurrently using the MapReduce technology.

References

1. STS Prasad and Anand Rajaraman *"Virtual Database Technology, XML, and the Evolution of the Web"*, Junglee Corporation,1998.

2. Wenhao Xu, Jing Li, *"VDM: Virtual Database Management for Distributed Databases and File Systems",* IEEE 2008 Seventh International Conference on Grid and Cooperative Computing, 978-0-7695-3449-7/08.

3. Ranieri Baraglia, Gianmarco De Francisci Morales, Claudio Lucchese, *"Document Similarity Self-Join with MapReduce"*, 2010 IEEE International Conference on Data Mining, 1550-4786/10

4. "Apache's Map/Reduce Tutorial" *http://hadoop.apache.org/docs/ r0.20.2.*

5. Ralf Lammel, "*Google's MapReduce programming model Revisited*", ELSEVIRE, Science Direct, Science of Computer Programming 70 (2008) 1–30.

6. "*What is MapReduce* " http://www.mapreduce.org/what-is-mapreduce.php.

7. Tom White "*Hadoop, The definitive guide"* ,ISBN 978-0-596-52197-4.

8. White Paper "*Deriving Deep Insights from big data analytics with map-reduce*" Teradata Corporation.