Enhancing Availability in Cloud Environment with Raid Based Map Reduce Approach

I.Bhuvaneshwarri
Assistant Professor (Senior), Department of Information Technology, Institute of Road and Transport Technology, Erode-638316, Tamilnadu, India

Abstract-One of the significant features of cloud computing is that computing is delivered via the Internet as services. Computing and IT resources are encapsulated as services, hiding all the details of implementation, deployment, maintenance and administration. Computing will be shifted from on premise systems to remote systems and users are connected to their data via the Internet. With cloud computing, deployment of IT systems and data storage is changed from on-premises user-owned IT infrastructures to off-premises third-party IT infrastructures. As the data is on the single cloud, data availability becomes a great challenge. Having the data on a single server scenario may breakdown at times, but in this work I split the big data and store that data under different server systems (multi server scenario), thus reduce the data loss and improve data availability. In this paper, I have introduced multi server scenario and use "Map Reduce" component of Hadoop with RAID level to improve the data storage and retrieval in cloud. It also increases the speed, performance, and integrity of the system.

Keywords: Cloud Computing; RAID; Map Reduce; Hadoop and multi server scenarios;

1. INTRODUCTION

The cloud computing infrastructure is revolutionizing all fundamental areas of IT -from security and investment in infrastructure to application development. Computing and IT resources are encapsulated as services, hiding all the details of implementation, deployment, maintenance and administration. With cloud computing, deployment of IT systems and data storage is changed from on-premises user-owned IT infrastructures to off-premises third-party IT infrastructures. The traditional single server system has many obstacles in data handling. In case of server down or failure, the whole data is lost and user cannot access their data at all. Big data also cause some burden to this system. However cloud storage systems must address challenges that are not addressed by traditional network or Distributed File Systems (DFS). Data storage and retrieval are two most important aspects of cloud services. Performing these in an effective way is the key building blocks for cloud computing applications based on the Map Reduce programming paradigm. In DFS, nodes (servers at different locations) simultaneously serve computing and storage functions; a data file is partitioned into a number of chunks allocated in distinct nodes so that Map Reduce tasks can be performed in parallel over on these nodes. In this research work will be concentrated on the above mentioned issues and provide enhanced performance and solutions based technique on RAID levels. In this paper Section – II discuss about single server scenario and multiple server scenario, Section –III discuss about proposed system, Section –IV covers problem definition, Section-V describes the contribution of this work and in Section – VI conclude the paper.

2. SINGLE SERVER VERSUS MULTI SERVER SCENARIO

A traditional server is a huge dedicated centralized computer that is used by all other users on the network to access its services. It has a number of different services from simple mail server, database server, web server to more complex real time server, resource server, etc. Hence it runs a more stable server operating system and is usually installed on "server grade" hardware, with many redundancies built in to improve reliability. Even though the server with redundant power supplies, hard drives and other components, there is a still large issue that leads to failure that is single point of failure, such as the mother board. So if that server has a hardware fault, software issue or any other power problem all of the users are off line and are out of services. Traditional servers require man power (humans) and hours (more often days) to recover from those failures and to launch again. Apart from single point of failure it also has server overload issues. Servers are only capable of executing a certain amount of commands per second, so extreme network congestion can cause the server to overload and crash, making files and resources unavailable to users. The disadvantages of traditional single server system are: Single point of failure, usually not suitable for frequent multitasking or for applications that require more CPU power and extremely busy networks can sometimes overload and crash a server. But in this paper, split the big data and store that data under different server systems (multi server scenario).It overcomes the drawbacks with single server scenario. Thus reduce the data loss and improve data availability.

3. PROPOSED SYSTEM

Parallel processing divides a large task into many smaller tasks, and executes the smaller tasks concurrently on several nodes. As a result, the larger task completes more quickly. Parallel processing is much faster than sequential processing when it comes to doing repetitive calculations on vast amounts of data. This is because a parallel processor is capable of multithreading on a large scale, and can therefore simultaneously process several streams of data. In this proposed system, I have introduced the concept of multi server scenario to overcome the drawbacks with single server. For this work, I have implemented Hadoop's Map Reduce Programming model with RAID-5 to improve the data storage and retrieval in cloud. In this work, I have taken big data as input and use "Map Reduce" component of Hadoop which splits, processes the data in parallel, merges and finally gives us the reduced output data. Then I have split that data and stored it under different server systems using RAID-5.This multi server scenario concept overcome the drawbacks with single server. In this proposed system, I have implemented Hadoop's Map Reduce Programming model with RAID-5 to improve the data storage and retrieval in cloud. The advantages of using DFS in cloud are: It overcomes the single node failure as data are fragmented among multiple servers, improves availability, allows parallel processing of information, and minimizes the network collision, integrity of data increases with parity information in each disk.

4. PROBLEM DEFINITION

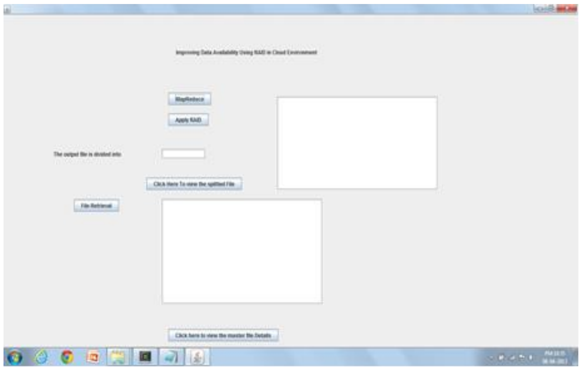
The paradigm shift of the industrial information technology towards a subscription based or pay-per-use service business model is known as cloud computing. The important aspect of quality of service in Cloud Computing relies on data availability, data security and data integrity. These aspects certainly pose new challenging issues for number of reasons. At first, traditional cryptographic primitives for the purpose of data security protection cannot be directly implemented due to the users' loss control of data under Cloud Computing. Considering various kinds of data for each user stored in the cloud and the demand of long term continuous assurance of their data safety, the problem of verifying accuracy of data storage in the cloud becomes even more challenging. Secondly, Cloud Computing is not just a third party data warehouse. The stored data in cloud may be frequently looking over by the users. To ensure data availability, storage integrity or correctness under dynamic data revise is hence of paramount importance. Most of the previous research works address data security issues in cloud under single server scenario and most of them do not consider data availability and data integrity (correctness) issues. For ensuring data availability and data integrity or storage correctness across multiple servers in a secured way, an effective scheme is required. Hence this research work will be concentrated on the above mentioned issues and provide enhanced performance and solutions based technique using RAID concept.

5. EXPERIMENTAL

In this system, the distributed data processing is achieved through the concept of MapReduce Programming model and distributed data storage is provided by HDFS (Hadoop DFS) which are the components of Hadoop. I have implemented MapReduce model to attain distributed parallel processing through JobTracker and TaskTracker components and to implement HDFS through NameNode and DataNode components. So the job execution starts when the client program submit to the

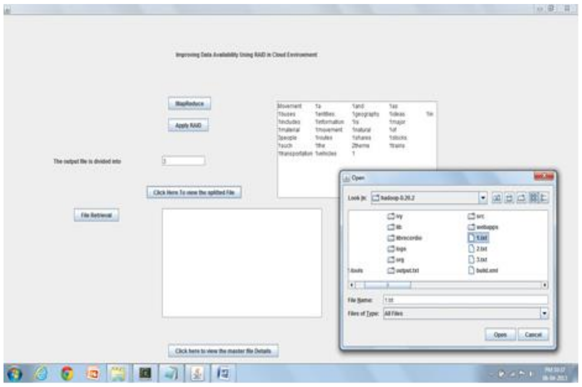
JobTracker a job configuration, which specifies the map, combine and reduce function, as well as the input and output path of the data. The JobTracker first consults the NameNode to determine the number of splits from the input path and select some TaskTracker based on their network proximity to the data sources and then the JobTracker send the task requests to those selected TaskTrackers. Each TaskTracker will start the map phase processing by extracting the input data from the splits. For each record parsed by the "InputFormat", it invokes the user provided "map" function, which emits a number of key/value pair in the memory buffer. A periodic wakeup process will sort the memory buffer into different reducer node by invoke the "combine" function. The key/value pairs are sorted into one of the R local files (suppose there are R reducer nodes). When the map task completes (all splits are done), the TaskTracker will notify the JobTracker. When all the TaskTrackers are done, the JobTracker will notify the selected TaskTrackers for the reduce phase. Each TaskTracker will read the region files remotely. It sorts the key/value pairs and for each key, it invokes the "reduce" function, which collects the key/aggregated Value into the output file (one per reducer node).The JobTracker keep tracks of the progress of each phases and periodically ping the TaskTracker for their health status. When any of the map phases TaskTracker crashes, the JobTracker will reassign the map task to a different TaskTracker node, which will rerun all the assigned splits. If the reduce phase TaskTracker crashes, the JobTracker will return the reduce at a different TaskTracker. RAID is a data storage scheme that uses multiple hard drives to replicate data among the drives. Depending on the configuration of the RAID or "RAID Level", the benefits of running RAID can be increased data availability, data integrity, fault-tolerance, throughput or capacity. And also RAID can improve system speed and can help prevent disk errors from compromising or corrupting data. Here also brought the concept of RAID 5 model in this system. RAID 5 is the most common secure RAID level. It is similar to RAID-3 except that the data chunks are transferred to disks by independent read and write operations (not in parallel). Striping combines several disk drives into a single volume. RAID 5 comprises block-level striping with distributed parity. Unlike RAID 4, where parity information is placed in a single disk, in RAID-5 instead of having a dedicated parity disk, parity information is spread across all the drives and if one disk fail and it won't affect any of the data. It requires that all drives but one be present to operate. Upon failure of a single drive, subsequent reads can be calculated from the distributed parity such that no data is lost. But RAID 5 requires at least three disks. It is a good all-round system that combines efficient storage with excellent security and decent performance. It is ideal for data (file) and application servers. I also simulated the components NameNode and DataNode of Hadoop. That is, the NameNode contains the details of the location of the data in the Data Node. Using the log details of the NameNode I can attain the user requested data which is stored across the many DataNode. This Input file has any text data like paragraph (Big data) and Output file has the resultant data which is spilt and stored at different DataNode using RAID 5. Also I have created Master node file which has the details about the data and in which DataNode it is present. The experiment evaluation is designed by using Cygwin which provides a LINUX based environment and Hadoop to implement MapReduce concept along with Java.

6. SAMPLE OUTPUT SCREEN SHOTS



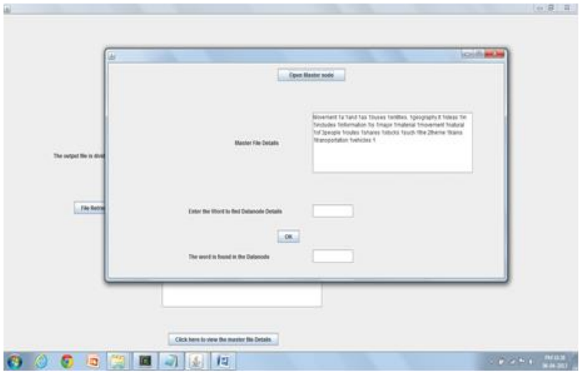
Improving Data Availability Using RAID in Cloud Environment

MapReduce
Apply RAID
The output file is divided into
Click Here To view the splitted File
File Retrieval
Click here to view the master file Details



Improving Data Availability Using RAID in Cloud Environment

The output file is divided into 3


Open Master node
Master File Details
The output file is divi
File Retri
Enter the Word to find Datanode Details
OK
The word is found in the Datanode
Click here to view the master file Details

Master File Details: Movement 1a 1and 1as 1buses 1entities, 1geography 1 1ideas 1in 1includes 1information 1is 1major 1material 1movement 1natural 1of 2people 1routes 1shares 1stocks 1such 1the 2theme 1trains 1transportation 1vehicles 1

7. CONCLUSION

I have effectively stored and retrieved data to and from the cloud environment in a secure and reliable manner. I have split the big data and storing them under multiple servers which reside at different locations to overcome the drawback of single server scenario. Then the data availability, integrity and security are also improved by applying the RAID 5 level of disk storage using the parity bits. Thus this system makes us to feel risk free storage and also more comfortable as it enhances fast retrieval of data.

REFERENCE

[1] Anil Gupta, Aaftab Qureshi, Parag Pande,"Cloud Computing Characteristics and Service Models:our own interpretation".

[2] J. Dean and S. Ghemawat. "Mapreduce: simplified data processing on large clusters". *Commun.ACM*, 51(1):107–113, 2008.

[3] Apache Hadoop, 2009. http://hadoop.apache.org/.

[4] M. Isard, M. Budiu, Y. Yu, A. Birrell, and D. Fetterly."Dryad: distributed data-parallel programs from sequential building blocks". In EuroSys '07: Proceedings of the 2nd ACM SIGOPS/EuroSys European Conference on Computer Systems 2007, pages 59–72, New York, NY, USA, 2007. ACM.

[5] Wikipedia, "www.wikipedia.org".