ISSN: 2348-3083
SJIF 2016 : 4.44
UGC APPROVED SR. NO. 48612
An International Peer Reviewed & Referred
SCHOLARLY RESEARCH JOURNAL FOR HUMANITY SCIENCE & ENGLISH LANGUAGE
SRJIS

LATEST DATA STORAGE TECHNIQUES IN CLOUD COMPUTING FOR DATA INTENSIVE APPLICATIONS

Ms. Musmade Anjali J.

Abstract

Cloud computing is becoming an emerging trend now days that provides many services like storage resources, computing and communication as a service over a network. Due to Communication resources there may be bottleneck while providing service for many data intensive cloud applications. For that purpose the effective solution is data replication that brings data nearer to data intensive applications. It helps in minimizing delays and high data availability. In this paper we study data replication strategies with considering number of aspects like Quality of Service, energy efficiency, load balancing in cloud computing data centers. Here we obtained the improved quality of service QoS with the help of all new strategies designed for data replication.

Keywords: *data replication, energy efficiency, Load balancing, cloud computing.*

1. Introduction

Day by day more development is going on in cloud computing system. So there is the major problem of big data storage and its management along with the security. For processing and storing this huge volume of data thousands of nodes are used. But doe to this large number of nodes in cloud computing there are many possibilities of hardware failure as well as software failure. This may result in data corruption due to that the running data-intensive applications unable to read data from disks successfully. To avoid this data corruption, the data replication technique is used in cloud computing system. It helps in providing high data availability to data intensive applications in cloud computing system. There are many new technologies are developing for data replications. Due to data corruption the QoS requirements of data intensive applications are not fulfilled successfully. The main aim of this paper is to investigate QADR problem and find out an optimal solution for it. The QADR problem is concerned about the QoS requirement of data intensive application. This problem is all about minimizing the data replication cost and also minimizing the QoS violated data replicas. To solve the QADR problem in this paper we are considering many different aspects along with the QoS requirement and for each aspect we want to design different algorithms. Along with the QoS requirements, we are interested to consider energy

consumption of the nodes as well as the energy efficiency requirements of data intensive applications for providing more throughput and high data availability for data applications developed in cloud computing system. After that we are also interested to maintain efficient load balancing amongst the number of data nodes for that purpose we are going to introduce new technology. Rest of the paper will describe the related work in second section, the system model and design in third section, system development and its performance is described in fourth section. Lastly the fifth section includes the conclusion of the paper.

2. Related Work

Early Work in data replication is all about how to store data amongst thousands of node connected through the network. For this number of techniques were introduced by many authors. Some of those techniques are considering QoS requirements; some of them are concerned about energy efficiency and later on some work is performed for load balancing amongst the number of nodes. We will discuss some of the related work in this section.

Author proposes two QoS-aware data replication (QADR) algorithms for cloud computing systems. One is high-QoS first-replication (HQFR) for performing data replication. and second algorithm which deals with minimum-cost maximum-flow (MCMF) problem. Author also proposes node combination techniques to reduce the probable large data replication time.[1]

The aim of author is to provide high data availability and to avoid data loss; for that purpose the best technique used in data storage is data replication. It allows minimizing network delays and bandwidth usage. Here author consider both energy efficiency and bandwidth consumption of the system. [2]

In today's generation; it is very important for the organizations to provide higher availability of quality services, computing resources and faster delivery. Author proposes an efficient load balancing algorithm for a Fog-Cloud based architecture. The algorithm uses data replication technique for maintaining data in Fog networks which reduces overall dependency on big data centers.[3]

The author find out the problem of how to place object replicas(e.g., web pages and images) to meet the QoS requirements of clients along with the objective of minimizing the replication cost. Author shows that the QoS-awareplacement problem for replica-aware services is NP-complete. Several heuristic algorithms for efficient computation of suboptimal solutions are proposed and experimentally evaluated. [4]

Author investigates the QoS-aware replica placement problem. Author proposes a new heuristic algorithm that determines the positions of replicas in order to satisfy the quality requirements imposed bydata requests.[5]

Author proposes Differentiated Replication (DiR), which allows users to choose different replication strategies by considering both the user requirements and system capability. Author implemented a system that offers four differentiated storage services with DiR. [7]

According to author, he has done the first attempt to study server failures and hardware repairs for large datacenters. They present a detailed analysis of failure characteristics as well as a preliminary analysis on failure predictors. They think that the results presented by them will serve as motivation to foster further research in this area.[8]

Here author researches on data replication in cloud computing data centers. They consider both aspects energy efficiency and bandwidth consumption of the system, along with the improved Quality of Service (QoS) as a result of the reduced communication delays. The main goal of the proposed replication strategy is to improve system performance while minimizing the energy consumption and bandwidth usage.[10]

A dynamic replacement strategy is proposed by author for a region based network where a weight of each replica is calculated to make the decision for replacement. The factors to be considered for replica replacement are size, cost, bandwidth and prediction of future access of the particular replica. Author proposed a region based framework. The design of the framework is based on the centralized data replication management.[12]

In this paper, a data replica selection optimization algorithm based on an ant colony system is proposed. The background application of the work is the Alpha Magnetic Spectrometer experiment, which involves large amounts of data being transferred, organized and stored. It is critical and challenging to be cost and time aware to manage the data and services in this intensive research environment.[13]

In this paper, authors propose a data replication strategy which adaptively selects the data files for replication in order to improve the overall reliability of the system and to meet the required quality of services. Further, the proposed strategy decides dynamically the number of replicas as well as the effective data nodes for replication. The popular data files are selected for replication based on employing a lightweight time-series technique, which analyzes the recent pattern of data files requests, and provides predictions for the future data requests.[16]

The given algorithms cannot solve QADR problem efficiently. In this QADR problem the main aim is to minimize total replication cost, provide high data availability, minimize the QoS-violated data replicas, minimum use of energy consumption by data nodes and equal load balancing of data amongst thousands of data nodes connected in cloud computing system. So according to our knowledge; in previous work all these aspects are not considered in the same system that means for each aspect there is a separate strategy or system was proposed by many authors. So for providing an efficient as well as optimal solution to QADR problem we are proposing this new strategy of data replication in cloud computing in data intensive application.

2. System Model

In this section we are discussing the model of our cloud computing system which we are using in our system. Here we are considering Hadoop Distributed file system (HDFS) architecture for the data storage system.

This is consists of name nodes, data nodes, switches and racks. Here is a single Name Node that manages the file system's metadata and namespace and multiple Data Nodes. The file content is divided into different blocks and each block of file is got replicated at different Data Nodes. Hence, Name Node is a Master node and Data Nodes are slave nodes; this as shown in fig. 1. Each Data Node has its own unique rack number.

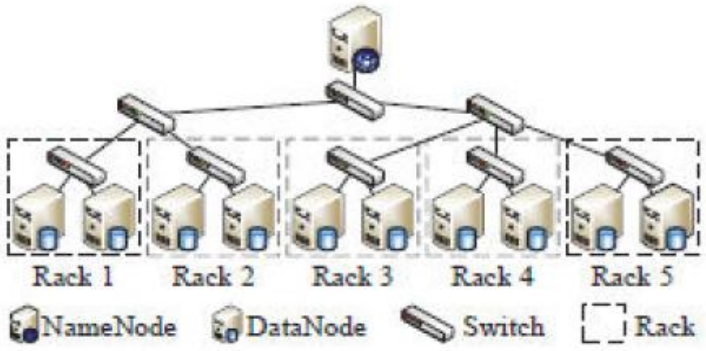

Fig. 1: Architecture of Hadoop Distributed File System

Now we will assume that at a time a single node can run one application. The file is divided into number of blocks each having size 64MB. We are maintaining two default data replicas against data corruption. One replica will be stored in the same rack and second will be stored in different rack. That is will overcome on the data corruption problem due to rack failure. While storing this data block or data replica the particular Data Node should be elected throughout the all Data Nodes available in Data Center of cloud computing. The Data Node should have high Quality of Service requirements. For this election we are designing a

new algorithm by taking idea from already existing HQFR (High QoS First Replication) algorithm. But this algorithm just satisfies only one objective of the QADR problem; to minimize data replication cost but what about minimizing QoS- violated data replicas. The solution is convert QADR problem into MCMF (Minimum cost Maximum Flow) Problem. After this we are designing an optimal solution for QADR problem known as Optimal Replica Placement algorithm. Until this we are achieving our two main objectives of QADR problem, but still energy consumption and load balancing is remaining.[1]

Here we are designing two separate strategies for both the remaining objectives. For energy consumption issue a new Energy Manager is introduced in this system. This manager will manage all the energy consumption requirements of the applications to fulfill their QoS requirements. This strategy optimizes energy consumption, network bandwidth and communication delay both between geographically distributed datacenters as well as inside each datacenter.[2]

There are very less work is done in the case of load balancing technologies developed in data replication for cloud computing system. After studying those articles we are designing a new strategy for balancing the load amongst number of nodes, so that it will maintain equal memory load as well as work load amongst those Data Nodes. It will minimize work overload as well as data storage overload among all the data nodes. And this will result in minimizing delay of the data requests executing by data intensive applications. Hence the QoS requirements of the applications are fulfilled. And the data will be available to the user very easily without failure.[3]

4.System Implementation

In this section, we will propose some efficient algorithms to solve QADR problem in data replication in the cloud computing system. The main goal is to minimize replication cost and to minimize QoS unsatisfied data replicas. We want to find an efficient algorithm for data replication with minimum energy consumption. Along with this also want to maintain load balance equally among the number of nodes.

For this purpose we are going to design a Replica Placement Manager (RPM) , a separate system which deals with all these problems in QADR. This RPM will be installed at the Name Node i.e. the Master node who is responsible for the storing of data blocks to particular Data Node according to the requirements. So all the applications which are sending the request for accessing the data from the cloud data will come to the Name Node first and then Name Node decides to which Data Node this request should be forwarded for further execution of the application. So our RPM will work here only. It will check all the aspects

like QoS requirements, energy consumption, and load balancing and then elect the qualified node for storing the data replica.

For that purpose we are developing some algorithms separately for each aspect inside our Replica Placement Manager. The first algorithm we are going to design is inspired from HQFR algorithm (High QoS First Replication). The second algorithm is inspired from MCMF (Minimum Cost Maximum Flow) algorithm. Then third one is designed for Energy Efficient Replication Algorithm. And the last one is Load Balancing algorithm.

4.1 HQFR algorithm: As per the name indicates High QoS First Replication algorithm serves the request first which is having higher QoS requirements. Here we are considering the QoS requirements from the aspect of request information and its access time only. The file which we want to store should be divided into blocks first and then these blocks are stored at appropriate location. In HDFS the data is divided into 64MB data blocks and the replication factor is two. That means other than original copy we are making two copies or replicas of data block. And those two copies are stored on different Data Nodes or different data racks. The Name Node keeps track of all the replicas other than original copy and they mounted on different data racks to avoid rack failure.

Basic idea of algorithm: As the name indicates the applications with high QoS should be replicated first. According to our knowledge the high QoS application have stricter requirements in the response time of a data access time than the normal applications. High QoS requirement application should take precedence over the low QoS requirement application to perform data replication. We have to sort all the applications according to their QoS requirement in a way, the application with high QoS should come first and then the lower one. If the data replication space is limited then first stores the data replicas of high QoS applications. When the high QoS application reads a corrupted data replica, its QoS requirements can be supported continuously by retrieving the data replica from high performance node. In the cloud computing system when any application performs a write operation then the node at which that application is executing will forward a replication request of a data block to the Name Node. The access time means the QoS requirement of that application is also attached with that request which going to generate a QoS aware replication request. Like this multiple QoS aware replication requests are issued in the cloud computing system from different nodes. But these requests are processed and sorted them in ascending order according to their associated access time. If the replication request r_i has higher QoS requirement than the replication request r_j that means the r_i has smaller access time than r_j. In such a case r_i will be processed first to store its data replicas in this algorithm.

While processing this replication request we have to find the qualified node's list; which help us to satisfy the QoS requirements of the appropriate application while running. The QoS requirement is given in the form of access time of that data block which is requested by an application. Note that while finding qualified node it should satisfy two conditions:

- The requested node R_i and its qualified node Q_j should not be mounted in the same rack. They should belong to two different racks.

$$Rack(R_i) \neq Rack(Q_j) \text{ ----------(1)}$$

Where Rack() is the function to determine in which rack a node is located.

- The total data replica access time from qualified node Q_j to request node R_i ($T_{access(Ri,Qj)}$)should be smaller than the QoS requirement of running application in R_i which is T_{qos}.

$$T_{access(Ri,Qj)} \leq T_{qos} \text{----------(2)}$$

After finding the qualified nodes by using these two conditions the data block can store its one data replica ineach qualified nodes and the qualified nodes update their replication space respectively. Now we will calculate the total replication cost. In HQFR algorithm the total replication cost is represented by the total storage cost taken by all the requested nodes to store their appropriate replicas. The replication cost is nothing but the total summation of storage costs of all data block replicas. But we are mainly interested in minimizing replication cost and also the number of QoS violated data replicas. For achieving second objective we are going to propose another algorithm for data replication.

4.2 An efficient replica placement algorithm: As its name indicates, this algorithm gives an efficient solution to the QoS aware replication problem. In this algorithm we are transforming QoS aware problem to the MCMF [Minimum Cost Maximum Flow] problem. As same to the previous algorithm in this algorithm also we are going to find out S_{qRi} the set of qualified nodes for each requested node R_i. Then after we will make union of the set of qualified nodes S_q with the newly derived set S_{qRi} which is set of qualified nodes corresponding to each requested node R_i. Then by using set S_r and S_q form a network flow graph. The vertices in the graph are from both the sets S_r and S_q and each edge represents the pair of appropriate capacity and cost of the data replication. Then by applying a suitable MCMF algorithm find out an efficient solution for that network flow graph. Then after we will perform the same operation for the unqualified nodes corresponding to each requested node R_i. Form the new graph from both the sets described above. Solve the graph by using same MCMF algorithm. Consider both the solutions obtained previously and perform an

efficient optimal placement of all QoS violated data replicas. Because of optimal placement of Qos-violateddata replicas the number of these replicas are minimized, which is our main goal. As we have used MCMF algorithm in this scheme, we get our solution in polynomial time. In this scheme the second part is having one flow graph. The amount of this flow graph is the amount of flow leaving from requested node R_i. Here we are considering the amount of flow leaving, which is not added to the total replication cost, which automatically helps in minimizing the total replication cost. Hence we achieved our both objectives. But we are interested inminimizing energy consumption in data replication. We are going to investigate another algorithm for energy optimization.

4.3 Energy Efficient Replication Algorithm: In this algorithm similar to above algorithm we are finding aset of qualified nodes corresponding to each requested node R_i. After that we willcheck status of each qualified node. So we will collect energy status of each node and make another set forthis nodes E_r. Then according to the energy status of nodes, sort them with higher energy node should come first. The replication request of that node should be considered first from the set of requested node. So the replication request is performed in minimum time with efficient energy.

4.4 Load Balancing Algorithm: In this algorithm we are checking the load status of the qualified nodes. And will elect the node with minimum load so that the access time will get minimized of the application. In this way the QoS requirement will get satisfied of the application.

So all these algorithms will run in Replica Placement Manager and after that will elect the appropriate qualified node for storing the data replica. This will help in achieving all our objectives and give optimal solution to the QADR problem.

5. Conclusion

Hence we found the optimal solution for the QADR (QoS Aware Data Replication) problem. We have designed a new system that will minimize the data replication cost as well as minimize the number of QoS-violated data replicas. Our system also works for minimizing energy consumption overhead as well as data overload amongst different numbers of Data Nodes. The main aim of the system is to satisfy Quality of Service requirements of the data intensive applications which are running in cloud computing system.

In future, we want to implement this system in real time cloud computing system and also want to calculate the execution time of each application running in cloud computing system.

References

J. W. Lin, C. H. Chen and J. M. Chang " QoS-Aware Data Replication for Data Intensive Applications in Cloud Computing Systems " in Digital Object Identifier 10, 1109/TCC 2013.

D. Boru, D. Kliazovich, F. Granelli, P. Bouvry, A. Y. Zomaya "Energy-Efficient data replication in cloud computing datacenters" in Spinger Science + Business Media New York 2015.

S. Varma, A. K. Yadav, D. Motwani, R. S. Raw, K. Sing "An Efficient Data Replication and Load Balancing Techniques for Fog Computing Environment" Preceedings of the 10th INDIAcom, IEEE conference ID:37465, 3rd International Conference on "Computing for Sustainable Global Development", March 2016 , ISSN 0973-7529, ISBN: 978-93-80544-20-5.

X. Tang, J. Xu "QoS-Aware Replica Placement for Content Distribution" IEEE Transaction on Parallel and Distributed Systems, Vol-16, No.10, Oct 2005.

H. Wang, P. Liu , J. Wu " A QoS-Aware Heuristic Algorithm for Replica Placement" Grid Computing Conference 2006.

K. Shvachko, H. Kuang, S. Radia, R. Chansler "The Hadoop Distributed File System" Storage Conference 2010.

Tung Nguyen, Anthony Cutway, and Weisong Shi "Differentiated Replication Strategy in Data Centers" IFIP International Federation for Information Processing 2010, NPC 2010, LNCS 6289, pp. 277–288.

Kashi Venkatesh Vishwanath and Nachiappan Nagappan "Characterizing Cloud Computing Hardware Reliability" SoCC'10, June 10–11, 2010, Indianapolis, Indiana, USA.

Rajkumar Buyya1,2, Saurabh Kumar Garg1, and Rodrigo N. Calheiros "SLA-Oriented Resource Provisioning for Cloud Computing: Challenges, Architecture, and Solutions" , 2011 International Conference on Cloud and Service Computing.

Dejene Boru, Dzmitry Kliazovich, Fabrizio Granelli, Pascal Bouvry, and Albert Y. Zomaya " Energy-Efficient Data Replication in Cloud Computing Datacenters" , Cloud Computing Systems, Networks, and Applications 2010.

G. Malathy, Rm. Somasundaram " Performance Enhancement in Cloud Computing using Reservation Cluster" European Journal of Scientific Research ISSN 1450-216X Vol. 86 No 3 September, 2012, pp.394-401.

K. Sashi and T. Santhanam " Replica Replacement Algorithm For Data Grid Environment" ARPN Journal of Engineering and Applied Sciences ISSN 1819-6608 , VOL. 8, NO. 2, FEBRUARY 2013

Wang, L., Luo, J., Shen, J. & Dong, F. "Cost and time aware ant colony algorithm for data replica in alpha magnetic spectrometer experiment" IEEE 2nd International Congress on Big Data (pp. 254-261). United States: IEEE 2013.

Dejene Boru, Dzmitry Kliazovich, Fabrizio Granelli, Pascal Bouvry, and Albert Y. Zomaya " Energy-Efficient Data Replication in Cloud Computing Datacenters" , Conference Paper in Cluster Computing , December 2013.ISSN: 978-1-4799-2851-4

S.Kirubakaran, Dr.S. Valarmathy And C.Kamalanathan, " Data Replication Using Modified D2rs In Cloud Computing For Performance Improvement", Journal Of Theoretical And Applied Information Technology, Issn: 1992-8645, 20th December 2013. Vol. 58 No.2.

Mohamed-K HUSSEIN, Mohamed-H MOUSA " A Light-weight Data Replication for Cloud Data Centers Environment" International Journal of Innovative Research in Computer and Communication Engineering, ISSN(Online): 2320-9801, Vol. 2, Issue 1, January 2014.

S. Soundharya "QADR with Energy Consumption for DIA in Cloud " International Journal of Computer Science and Mobile Computing, Vol.3 Issue.4, April- 2014, pg. 131-138.

Vijaya -Kumar-C "Optimization of Large Data in Cloud computing using Replication Methods" (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3) , 2014, 3034-3038.

Shipra Gupta, Indu Kashyap "Cost and Time Evaluation of Load Balancing and Service Broker Strategies in Multiple Data Centers" International Journal of Computer Applications (0975 – 8887) Volume 103 – No 17, October 2014.