Scholarly Research Journal for Humanity Science & English Language,
Online ISSN 2348-3083, SJ IMPACT FACTOR 2016 = 4.44, www.srjis.com
UGC Approved Sr. No.48612, FEB-MAR 2018, VOL- 6/26

IMPROVED REPLICATION AND LOAD BALANCING STRATEGIES IN CLOUD COMPUTING

Ms. Musmade Anjali J.

Abstract

In day today's life, Cloud computing is becoming very famous as a service provider among the several developers. It provides many services21` like storage resources, computing and communication as a service over a network. It is very suitable platform for developing new applications. The main aim of cloud computing is minimizing delays and provide high data availability. Since more and more applications are developing in cloud computing a major problem arises here is of big data management. While dealing with this big data many problems arises; here in this paper we study QADR problem in detail with considering number of aspects like Quality of Service, energy efficiency, load balancing in cloud computing data centers. We are proposing an optimal solution to this QADR problem with respect to QoS requirements as well as load balancing among different Data Nodes. For that purpose we are proposing a system to solve the QADR problem. In this system we are developing some algorithms for each aspect separately. So the appropriate Data Node is get selected through all these algorithms and data block replica is stored at this Data node.

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

1. Introduction

Cloud computing is becoming an emerging trend in IT industry for developing data intensive applications. Cloud computing is nothing but a service provider. It provides services like IaaS (Infrastructure as a Service), PaaS (Platform as a Service), SaaS (Software as a Service). Cloud computing have many features such as:- high flexibility, high security, easy to maintain, location independent, reduction in capital expenditure on hardware and software. Due to these features more and more data intensive applications are developing in cloud computing system. This results in a problem of maintaining huge volume of data in cloud computing system. As there are huge volume of data stored in cloud computing system there may be possibilities of software or hardware failure at data centers and due to that there may be possibilities of data corruption; that indirectly violate the QoS (Quality of Service) requirements of the applications. This problem is known as QADR (QoS Aware Data Replication) problem.[1]

In this paper we want to investigate this QADR problem and want to find out an efficient solution to this QADR problem. To avoid this kind of data corruption problem, the data replication strategy is used at several data centers. Many more research is going on in data replication strategy now days. Data replication means maintaining duplicate copies of original data so that if original data is get lost by any problem then the duplicate that means replica can be used of that, and in this way the data corruption is get tolerate. If such a corrupted data is requested by any application for the access and that data is not available for the application will result in violation of Quality of Service of that application. This is nothing but the problem of QADR. To solve this problem we are proposing a new system. This system deals with the QoS requirements of the application, minimization of total replication cost and also minimizes the QoS violated data replicas. In addition to this system is also going to consider energy consumption requirements and load balancing problem within the Data Nodes of Cloud computing data centers. Remaining paper is distributed in following sections: the related work in second section, the system model and design in third section, system development and its performance in fourth section, conclusion of the paper in the fifth section.

2. Related Work

Here we will discuss some existing technologies developed in the same scenario by different authors. To tolerate the data corruption problem, data replication technology is used in cloud computing system to provide data availability. But here Qos requirements of the application are not considered. In this paper, the QADR problem is investigated firstly. The QADR problem concerns how to efficiently consider the QoS requirements of applications in the data replication. To solve the QADR problem, author proposes a greedy algorithm, High QoS first replication (HQFR) algorithm. In second algorithm the QADR problem is transform to MCMF (Minimum cost Maximum flow) algorithm. However these two algorithms run in polynomial time. [1]

Here author proposes a data replication technique for cloud computing data centers which optimizes energy consumption, network bandwidth and communication delay both between geographically distributed data centers as well as inside each datacenter. The system focuses on the performance of cloud applications, utilization of communication resources and energy efficiency.[2]

In this paper author uses fog server system for balancing load in the network and the cloud servers. Author proposes three tier architecture; including cloud server tier, fog server tier and ground tier. The fog server functions as a middleware. The roles for administration of server process, IP packets, load and communication are optimized by Fog Server Manager(FSM). Each fog server carries one FSM.[3]

In this paper, author investigated the QoS aware replica placement problem for responsiveness QoS requirements. They consider two classes of service models: replica-aware services and replica-blind services. In the replica-aware services, the servers are aware of the locations of replicas and can therefore optimize request routing to improve responsiveness. In replica-blind services, the servers are not aware of the location of the replicas or even their existence. Efficient algorithms are proposed to compute the optimal locations of replicas under different cost models.[4]

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

Here author studied server failures and hardware repairs for large data centers. They present a detailed analysis of failure characteristics as well as a preliminary analysis on failure predictors. Author characterizes server failures for large data centers.[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 calculatedto 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 thecentralized data replication management.[12]

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

In this paper, authors propose a data replication strategy which adaptivelyselects the data files for replication in order to improve the overall reliability of the system and to meet the required qualityof services. Further, the proposed strategy decides dynamically the number of replicas as well as the effective data nodesfor 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]

All the above given algorithms unable to 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. As far as our knowledge is considered; in previous work all these aspects are not considered in a single system that means for each aspect there is a separate strategy or system was proposed by many authors. So for providing an optimal solution to QADR problem we are proposing the new strategy of data replication in cloud computing for data intensive application.

3. System Model

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

This architecture includes NameNodes, DataNodes, Switches and Racks. Here is a single NameNode that manages the file system's metadata and namespace and multiple DataNodes. The file content is divided into different blocks. And then these blocks are stored at DataNodes. This distribution is done according to the aspect of security.While performing data replication; each block of fileis replicated at different DataNodes. Here NameNode is a Master node and DataNodes are slave nodes; this is shown in fig. 1. There are number of racks of DataNodes in each DataCenter. Each rack has its own unique rack number. Each

DataNode has its own Node number as well as a rack number;in which rack that DataNode is placed. All these numbering is maintained by the NameNode. NameNode have to keep track of each data block, where its original copy is stored or where its replicas are stored.

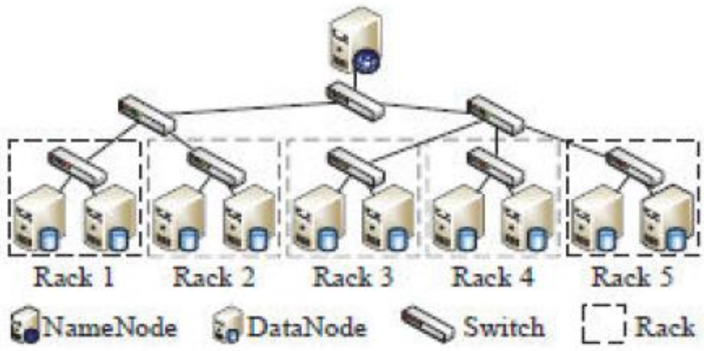

Fig. 1: Architecture of Hadoop Distributed File System

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 will overcome on the data corruption problem due to rack failure. While storing this data block or data replica the particular DataNode should be elected throughout the all DataNodes available in Data Center of cloud computing. The DataNode 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.[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 is no sufficient work is done regarding the issue of load balancing in data replication of 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 number of DataNodes present at DataCenters. It will minimize work overload as well as memory overload among all the datanodes. 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 made available to the user very easily without failure.[3]

As far as our knowledge is concerned there is no such a system which deals with all the aspects like QoS requirements, energy consumption and load balancing at the same time. In previous work there is a separate strategy for each aspect. So our aim is to design such a system which deals with all the aspects at the same time. We are going to elect an appropriate DataNode through all the algorithms and then replicas are stored at appropriate DataNode. This will minimize the replication cost and minimize the number of QoS-violated data replicas, also minimizes the energy consumption and the load overhead.

4. System Implementation

Here, we are proposing some algorithms to solve QADR problem in data replication of the cloud computing system. The main goal of our system is to minimize replication cost and to minimize QoS unsatisfied data replicas. We also want to find an efficient algorithm for data replication with minimum energy consumption and for maintaining load balance equally over the number of nodes.

Weare going to design a Replica Placement Manager (RPM), a system which deals with all the problems of QADR problem. This RPM should be installed at the NameNode i.e. the Master node; which is responsible for managing and updating of data blocks at different DataNodes in the cloud computing system. The request is send by any application for data access is come to the NameNode first, and then NameNode will decide to which DataNode the request should be forwarded for further execution. 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.

Weare developing some algorithms in the same system RPM but separately for each aspect. 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. And the last one is for Load Balancing.

4.1 HQFR algorithm: According to the name High QoS First Replication algorithm serves the request first which is having highest QoS requirements. Here the QoS requirementsare considered from the aspect of request information and its access time only. The file which is going to store in cloud computing data centers; should be divided into blocks first and then these data blocks are stored at appropriate location in DataNodes. InHDFS (Hadoop Distributed File System) the data file is divided into 64MB data blocks and the replication factor is two by default. Weare making twocopies or replicas of data block different from the original copy of data block. These two copies are stored on differentDataNodes or different data racks. The NameNode keeps track of all the replicas other than original copyand they mounted on different data racks to avoid rack failure.

According to our knowledge the high QoS application have stricter requirements in the response time of adata access time than the normal applications. High QoS requirement application should take precedence overthe 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 thelower 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 are placation request of a data block to the NameNode. 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 helps us to satisfy the QoS requirements of the appropriate application while running. The QoS requirement is given in the form ofaccess time of that data block which is requested by an application.

The algorithm is described as below:-

- Input: The list of all the connected nodes in the cloud to the master node.
- Output: The list of qualified nodes.
- Step 1: create a list of all the nodes connected to the master node R.
- Step 2: Get access time of each node from the node list R.
- Step 3: Sort the node list R in ascending order in a way that node having minimum access time (high Qos) will appear first.
- Step 4: Store the sorted list of nodes in qualified node list Q.

Note that while finding qualified node itshould 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 willcalculate the total replication cost. In HQFR algorithm the total replication cost is represented by the totalstorage cost taken by all the requested nodes to store their appropriate replicas. The replication cost is nothingbut the total summation of storage costs of all data block replicas. But we are mainly interested in minimizingreplication cost and also the number of QoS violated data replicas. For achieving second objective we aregoing to propose another algorithm for data replication.

4.2 An efficient replica placement algorithm: This algorithm gives an efficientsolution to the QoS aware replication problem. In this algorithm we are transforming QoS aware problem tothe 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 MCM Falgorithm 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 $R_{i.}$ The algorithm is described as follows:-

- Input: The list of all the connected nodes in the cloud to the master node.
- Output: Data replicas are placed to the qualified nodes and removal of unused QoS violated data replicas.
- Step 1: Create a Graph G =(V,E) by using the list of nodes connected to the master node R in a way such that master node is a source and qualified nodes are sink.
- Step 2: Create a two dimensional matrix and stores the cost of replication and capacity of data flow from one datanode to other.
- Step 3: Apply MCMF algorithm to this network graph and get efficient solution having minimum cost and maximum flow.
- Step 4: Now apply Replication Policy to the elected nodes derived from MCMF algorithm.
- Step 5: Replicas are get stored to their appropriate location and removal of excess QoS violated data replicas.

Here we areconsidering the amount of flow leaving, which is not added to the total replication cost, which automaticallyhelps 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 energyoptimization.

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 will check status of each qualified node. So we will collect energy status of each node and make another set for this 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. The algorithm is described as follows:-

- Input: The list of all the connected nodes in the cloud to the master node.
- Output: The list of qualified nodes.
- Step 1: create a list of all the nodes connected to the master node R.
- Step 2: Get energy status of each node from the node list R and set energy flag.
- Step 3: Sort the node list R in descending order in a way that node having higher energy status will appear first.
- Step 4: Store the sorted list of nodes in qualified node list QE.

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. The algorithm is given as follow:

- Input: The list of all the connected nodes in the cloud to the master node.
- Output: The list of qualified nodes.
- Step 1: create a list of all the nodes connected to the master node R.
- Step 2: Get memory storage information of each node from the node list R.
- Step 3: Sort the node list R in ascending order in a way that node having higher memory storage area will appear first.
- Step 4: Store the sorted list of nodes in qualified node list QL.

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

5. Result Analysis

We have created our own cloud computing environment as per the possible levels. We are successful in passing messages between different nodes in the cloud computing system. Now we are applying our applications to the data nodes through the name nodes. We are performing data replication by using our proposed algorithms and trying to sort out our problem. According to the time complexities of the first three algorithms we have got some observations as per the number of node considered. It is represented as below in chart:-

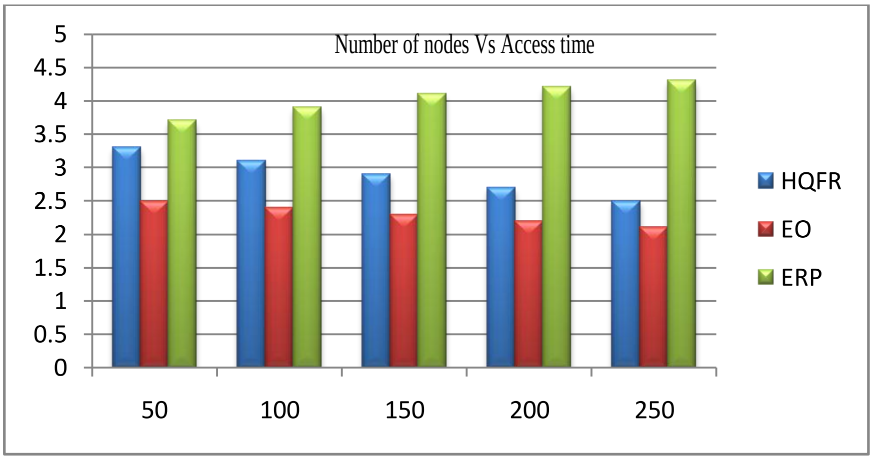

Fig.5.1 graph for number of nodes Vs access time

We also have done comparison with existing system in case of QoS violated replicas and the result is shown below in the form of graph:-

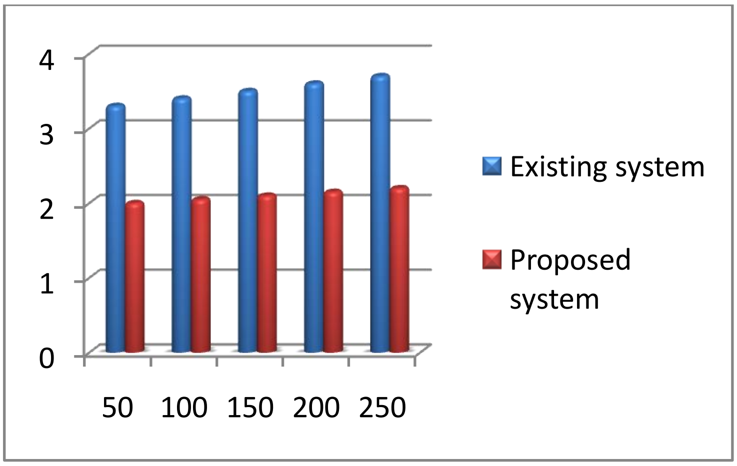

Fig.5.2 graph for Number of QoS violated data replicas Vs number of nodes

6. Conclusion

After investigation we find out the problem of QADR (QoS Aware Data Replication) in case of data replication in cloud computing systemfor data intensive applications and then develop an optimal solution for the QADR problem. This new system helps to minimize the total data replication cost and also to minimize the number of QoS-violated data replicas. This system also works for minimizing energy consumption overhead and data overload amongst different numbers of DataNodes. 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. We want to provide high data availability in minimum access time. As far as our knowledge is considered very few systems are working in this scenario. So we developed a single system; which works for all the aspects like QoS requirements, energy consumption, and memory overload. By applying all these algorithms we find the appropriate DataNode for data replication.

In future, we want to implement this system in real time cloud computing system and also want to find out the rate of execution time for 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.

M.Amutha, R.Sugumar "Comparative Analysis on Data Replication System in Clouds Based On Data Mining Techniques", International Journal of Innovative Research in Computer and Communication Engineering, ISSN(Online): 2320-9801, Vol. 3, Issue 8, August 2015.

Rashmi Ranjana T P, Jayalakshmi D S, Dr.Shrinivasan R "On Replication Strategies for Data Intensive Cloud Applications"(IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (3) , 2015, 2479-2484

Dr. Sunita Varma, Ms. Gopi Khatri "QoS-Aware Data Replication in Hadoop Distributed File System" Int. J. Advanced Networking and Applications Volume: 07 Issue: 03 Pages: 2741-2751 (2015) ISSN: 0975-0290

Manisha Kalkal*, Sona Malhotra "Replication for Improving Availability & Balancing Load in Cloud Data Centres" International Journal of Advanced Research in Computer Science and Software Engineering 5(4), April- 2015, pp. 108-110.

Pradeep Singh Rawat, Priti Dimri, G.P. Saroha, Varun Barthwal "A Load Balancing Analysis of Cloud Base Application with different Service Broker Policies" International Journal of Computer Applications (0975 – 8887) Volume 135 – No.10, February 2016

Loheswaran.K1, Anbarasu.S "Replication Approach In Cloud Computing On Data Storage Security" International Journal Of Scientific Engineering And Applied Science (Ijseas) – Volume-2, Issue-3, March 2016 ISSN: 2395-3470