ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

MODELING FUZZY BASED REPLICATION STRATEGY TO IMPROVE DATA AVAILABIITY IN CLOUD DATA CENTER

Nivetha N K[1] , Vijayakumar D[2]
PG Scholar [1] , Assistant Professor(SG)[2]
National Engineering College, Kovilpatti, India.
nivethakrishnasamy@gmail.com [1] , academic.dvk@gmail.com[2]

.

***Abstract* -- Data management is an important strategy to be considered in large scale cloud environments. This can be easily handled by using file replication. This project is to improve data availability in cloud data center by taking replicas and place them into different data centers. Replica creation may depends on the maximum usability / accessibility of the data in cloud data center. The data which accessed most frequently are consider for replication process. Replica selection is to select most suitable replica, in order to place them into more than one data centers.**

Index terms-- Cloud computing, Data center, replication.

I.INTRODUCTION

Cloud computing is a model for shared pool of resources with on-demand access. Cloud computing is an internet based computing which provides shared pool of resources which can be accessed anywhere and anytime all over the world and offered on-demand to a user who doesn't need to bother with implementation details or maintenance. It involves delivering hosted services to the user. These services are broadly categorized into Infrastructure-as-a–Service, Platform-as-a-Service and Software-as-a-Service. Data management is the major issue in cloud computing. An efficient data management provides high data availability in cloud data centers. The proposed method includes three main categories. First part includes replica creation based on the availability of particular data. In second part, replica selection process (selection of best replica) takes place based on the network performance. Third part includes the placing and replacing of replica into data centers. The final step includes replica selection process using Fuzzy based replica selection, selection procedures are configured to select the optimum number of replicas. The data availability can increase by this replication method.

II.DATA REPLICATION

Data replication is one of the way to supporting data integration requirements. Data replication provides data synchronization and availability, enables efficient data growth while increasing revenue using up-to-the-minute information. Real time analytical data synchronization enriches many big data systems and mobile applications, improve the event-driven business. Robust data replication capabilities deliver large volumes of data with very low latency, making data replication ideal for multi-site workload distribution and continuous data availability. Figure 2.1 shows the basic data replication strategy, where the source data were replicated and placed in three different targets.

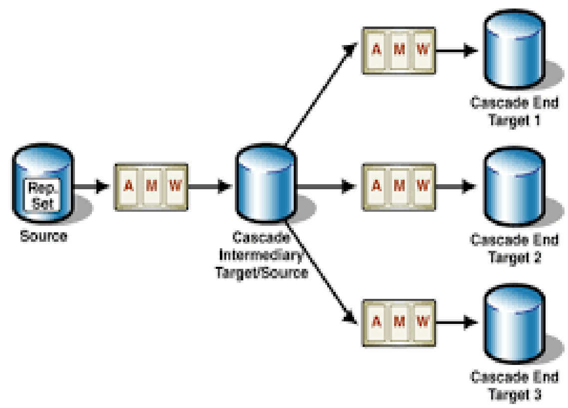


Figure 2.1 Data Replication

Data replication process used to maintain multiple copies of data called replicas. Replication improves the availability of data by allowing access to the data even some of the replicas are unavailable. It also improves system performance through reduced latency, by allow users to access nearby replicas and avoiding remote network access..

III.RELATED WORKS

Cloud computing faces major issues in Data Management strategies. Replication is an active area of research for many years. Data replication method classified into two types. One is static replication and the another one is dynamic replication. In static replication, the number of host and number of replicas are pre-determined. But in case of dynamic replication, it can automatically creating an deleting

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

replicas based on data accessibility .

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

Bakhta meroufel et al (2012) [1] proposed an approach for dynamic replication in a hierarchical grid that takes into account crash failure in the system. Dynamic replication is based on two parameters of data management is availability and popularity. The percentage of availability may increase depends upon the popularity of the data. This approach provides to assure the availability even in the presence of failures.

Najme Mansouri et al (2012) [6] proposed DHR algorithm (Dynamic Hierarchical Replication algorithm) that has three parts like replica selection, replica decision and replica replacement DHR algorithm effectively reduce the file access time due to the limited storage space of Grid sites. DHR deletes those file that exist in local LAN (i.e. files with minimum transfer time) when free space is not enough for the new replica. It stores the replicas in the best site where the file has been accessed most, instead of storing files in many sites.

Reena S. More et al (2015) [7] proposed the custom partitioning algorithm. The custom partitioning algorithm's basic idea is to partition a large problem into smaller sub problems. The proposed system replicate the data on the high performance node and reduce the energy by processing the data on high performance node and also reduce the replication cost.

Sai-Qin Long et al (2014) [8] proposed a multi-objective offline optimization approach for efficient replica management. MORM(Multi-objective Optimized Replication Management) make decision about replication factor and replication layout based on auto immune algorithm. MORM scheme considers the historical information and it is fed into the system. Thus the MORM replication management with the intent of selecting replication factor and replication layout strategy for files that optimizes all the major objectives. Thereby the number of replicas of highly replicated file was handled using MORM replica strategy.

IV.PROPOSED WORK

Main goal of this project is to increase the data availability by holding replicas in different Data centers and to decrease energy consumption by limit the number of replication copies based on frequent usage of resource. When a primary Data center crashed due to any reason, then user request redirected into next available zones. The energy consumption reduced by using our algorithm which based on fuzzy logic. Here we refer MORM (Multi objective Optimized Replication Management) algorithm to increase data availability in cloud data centers. ***System Architecture***

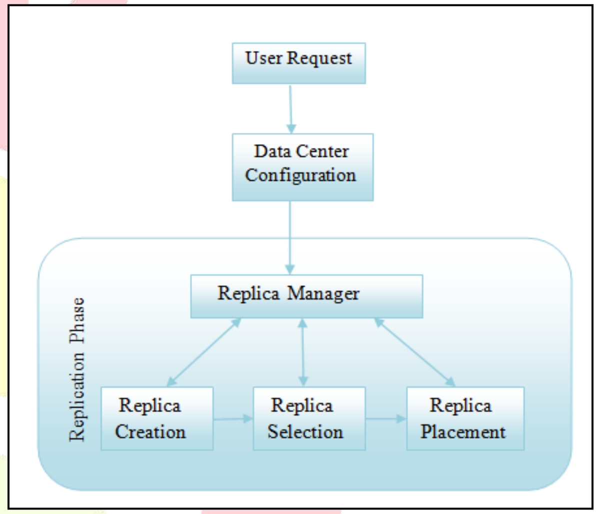


Figure 4.1. Architectural diagram of proposed system

The proposed architecture is composed of four modules. The modules are:

1. Data center configuration
2. Replica creation
3. Replica selection
4. Replica placement

4.1.1. Data Center Configuration

In data center configuration module, the basic details like data center name, data center region and number of virtual machines are configured by the cloud owner / authorized user. The cost settings are based on the service provided to the customer. Which may depends on the type of user.

4.1.2. Replica Manager

Replica manager handle entire operations of replication phase. Which includes,

1. Replica creation
2. Replica selection
3. Replica placement.

Replica manager includes all the metadata about the replicas like location, size and the number of replicas. Replica manager redirects the user request to another available Data center / zone while the primary Data center crashed or failed.

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

4.1.2.1. Replica Creation

In replica creation phase, based on the number of user requests, the replication takes place. Replica creation phase creates the number of extra copies and place them into a different data centers. Maximum of three replication copies are allowed at the time in many popular file systems such as Google File System (GFS) and Hadoop Distributed File System (HDFS).

When the replica placed in primary Data center, then the file will be directly accessible to the user. If the user requesting file is not available in primary Data center(primary available zone), then it automatically redirected to another data center, which holds replica copy of same resource. If the object is not available in primary Data center, and if it has enough capability to store new copy of original resource, then the replica will created and placed in the primary Data center.

4.3.1.2. Replica Selection

Replica selection phase involves selection of optimum replica among all the data centers (like primary, secondary zones)and selecting the optimum replica of the object for provisioning during its demand. The optimal replica selection causes the network utilization with lower bandwidth consumption. Replica selection is entirely based on the capability and performance of the available data center. When the resource of particular data center crashed or failed due to natural disasters, then the user request processed by the secondary available zone. This increases the fault tolerance of the system. The replica selection includes the optimum selection of data for user request, which may helpful to the faster data processing in cloud data center as well as the low memory consumption. Two cases to be considered during replica selection:

1. Select replica from nearest data center in order to reduce the bandwidth.
2. Select zone with minimum number of user request, which reduces the network congestion.

4.1.2.3. Replica Placement

In replica placement phase, the newly created replicas are placed in the suitable Data center. This increases the data availability due to place the same resource (replica of original resource) in more than one data center. In order to reduce storage capacity, the replicas are placed in new Data center which does not have that resource already, and the replica updation process frequently takes place after the placement.

V. TOOL ANALYSIS

Cloud analyst: Cloud analyst is a cloud simulator that is completely GUI based and supports the evaluation of social network tools according to geographic distribution of users and data centers.

There are three service broker policies

a. Closest data center connection
b. Optimize response time
c. Re-configure dynamically

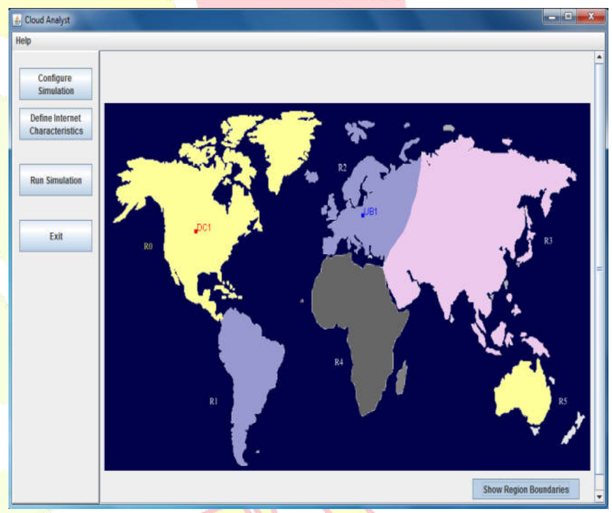


Figure 5.1 Cloud analyst screenshot

Figure 5.1 shows the cloud analyst tool design and the different colors shows different regions. Six regions are represented here. For each region, it has one or more user bases and that are having connection with data centers that are placed in unknown region.

5.1. Comparison between service broker policies:

Comparison between the scenarios
i) Closest Data Center
ii) Optimize response time
iii) Reconfigure Dynamically

Scenario-1: 10 User Bases & 1 Data Center

Data center region : 3
No.of data centers : 1
No.of userbase : 10

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

	Closest data center	Optimize response time	Reconfigure dynamically
Overall response time	Avg (ms) - 571.23 Min (ms) - 38.86 Max (ms) -1245.12	Avg (ms) - 570.21 Min (ms) - 39.36 Max (ms) - 1260.12	Avg (ms) - 571.70 Min (ms) - 39.11 Max (ms) - 1245.61
Data center processing time	Avg (ms) - 0.27 Min (ms) - 0.02 Max (ms) - 0.86	Avg (ms) - 0.27 Min (ms) - 0.01 Max (ms) - 0.87	Avg (ms) - 0.75 Min (ms) - 0.02 Max (ms) – 30.01
Data center request servicing times	Avg (ms) - 0.27 Min (ms) - 0.02 Max (ms) - 0.86	Avg (ms) - 0.27 Min (ms) - 0.01 Max (ms) - 0.87	Avg (ms) - 0.75 Min (ms) - 0.02 Max (ms) – 30.01
Total virtual machine cost ($)	0.50	0.50	3.26
Total data transfer cost ($):	0.64	0.64	0.64
Grand total: ($)	1.14	1.14	3.90

Table 1- Comparison between three scenarios for 10 UB & 1DC

Scenario 2: 50 User Bases & 1 Data Center

Data center region : 3
No.of data centers : 1
No.of userbase : 50

	Closest data center	Optimize response time	Reconfigure dynamically
Overall response time	Avg (ms) - 647.14 Min (ms) - 37.79 Max (ms) - 1300.12	Avg (ms) - 647.48 Min (ms) - 38.61 Max (ms) - 1305.11	Avg (ms) - 647.78 Min (ms) – 38.41 Max (ms) - 2680.01
Data center processing time	Avg (ms) - 0.25 Min (ms) - 0.01 Max (ms) - 0.90	Avg (ms) - 0.25 Min (ms) - 0.01 Max (ms) - 0.90	Avg (ms) - 0.86 Min (ms) - 0.01 Max (ms) – 1635.01
Data center request servicing times	Avg (ms) - 0.25 Min (ms) - 0.01 Max (ms) - 0.90	Avg (ms) - 0.25 Min (ms) - 0.01 Max (ms) - 0.90	Avg (ms) - 0.86 Min (ms) - 0.01 Max (ms) – 1635.01
Total virtual machine cost ($)	0.50	0.50	3.25
Total data transfer cost ($):	3.20	3.20	3.20
Grand total: ($)	3.70	3.70	6.46

Table 2- Comparison between three scenarios for 50 UB & 1 DC

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special issue 15, March 2016

The two service broker policies i) closest data center and ii)optimize response time are more or less same in all aspects like response time, processing time, data transfer cost, etc,. When compare with reconfigure dynamically policy, other two are better in performance as well as in cost aspects.

VI. PROPOSED ALGORITHM

Algorithm : Replica placement

Input: Integer r, the number of replicas
Input: Integer dc, the number of datacenters

Output: Replicas [0...dc - 1][0...r -1]

```
for all i such that i = 0 or i < 3 do
//(i – maximum allowable count of replicas. Where i=3)
{
        Preload node dc's replicas with dc
}
for all i such that i < 3 do
{
        repeat until i=3
        z = random node, s.t. 0 = z < dc
        v = Rep[z]
        until z = i
        if v = i and Rep[i] = z then
        {
                valid rep = 1
                if Rep[i]== v or Rep[i]== Rep[z] then
                {
                        valid rep = 0
                }
                if valid rep then
                {
                        Rep[z]= Rep[i]
                        Rep[i] = v
                }
        }
}
```

The above algorithm describes about replication placement algorithm. Here, inputs are r (Number of replicas) and dc (Number of data centers) and the output is number of replicas placed in number of data centers.

VII. IMPLEMENTATION RESULTS

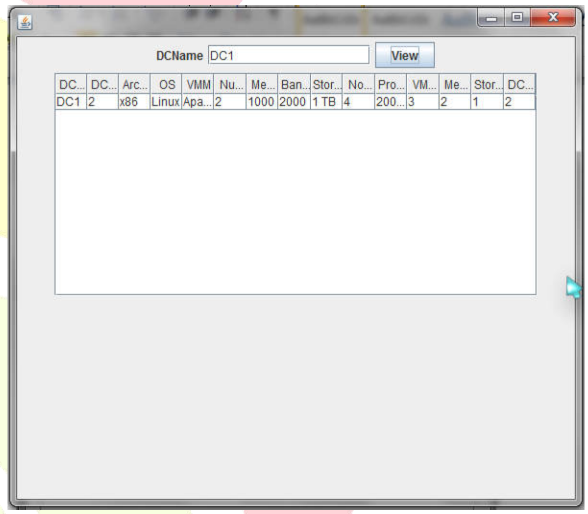
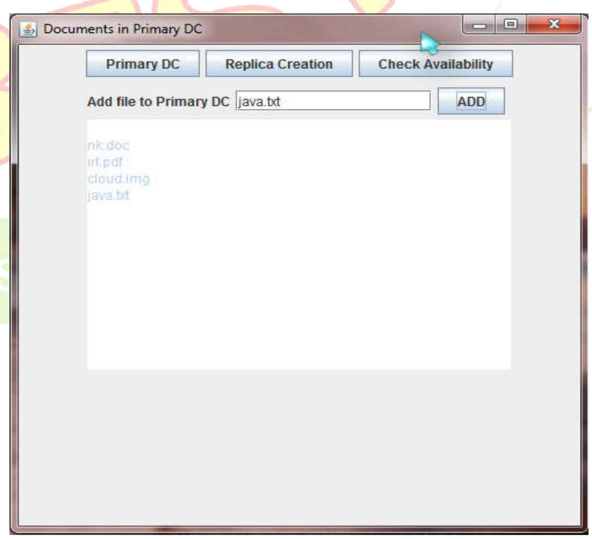


Figure 7.1 Screen shot for displayed DC details

Figure 7.1 shows, the retrieved details of the particular data center. The cloud service provider can insert or modify data from the cloud. These features only accessible by the cloud service provider.



ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

If the replica count is equal to zero or less than 3, then preload the original resource for initialize replication process. Select node z (random node) to place newly created replica and set the current replica count (v) is equal to number

of datacenter have that resource. Repeat this steps until the count reaches 3. If the newly created replica exactly match with the original resource, then that is called valid replica (i.e valid rep=1) otherwise, invalid (i.e, valid rep=0).

Figure 7.2 Screen shot for DC documents

Figure 7.2 shows, the available documents in primary data center. These data are handled by the user and managed y the service provider.

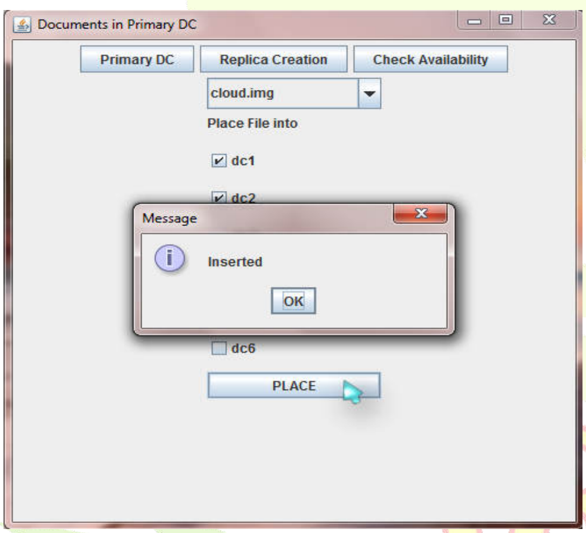


Figure 7.3 Screen shot for replica creation

Figure 7.3 shows the replica creation method. The selected file (cloud.img) placed in two different data centers DC1 and DC2.

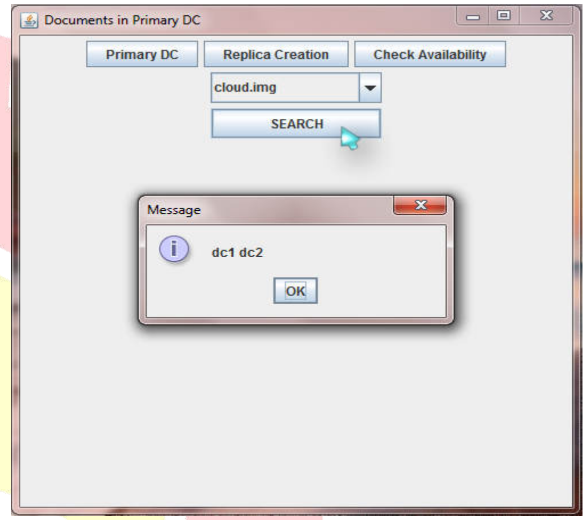


Figure 7.4 Screen shot for checking availability

In Figure 7.4, the file availability is checked. The file "cloud.img" is place in two data centers DC1 and DC2.

The above implementation includes data center configuration and replica creation modules. For data center configuration, the service provider should set up the configurations. For replica creation, it depends on the frequent usage of same file in cloud data center, and which make the copy of file is known replica creation.

VIII. IMPLEMENTATION IN TOOL (CLOUD ANALYST)

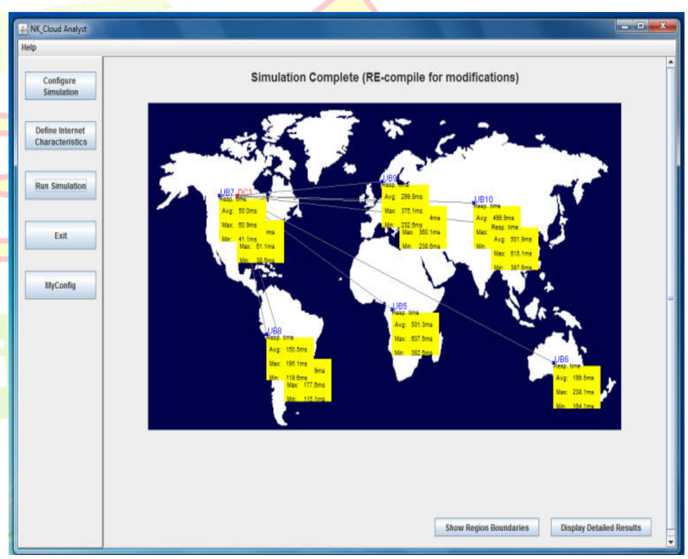
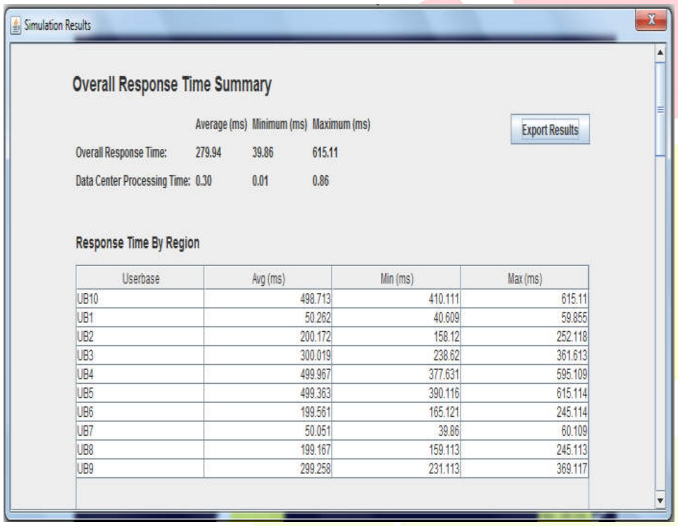


Figure 8.1 Simulation with Single Data Center & 10 User Bases

Figure 8.1 shows the simulation result with single data center

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

and 10 user bases. Because of the single data center concept, the replication concept can't achieved here.


Simulation Results

Overall Response Time Summary

	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	279.94	39.86	615.11
Data Center Processing Time:	0.30	0.01	0.86

Export Results

Response Time By Region

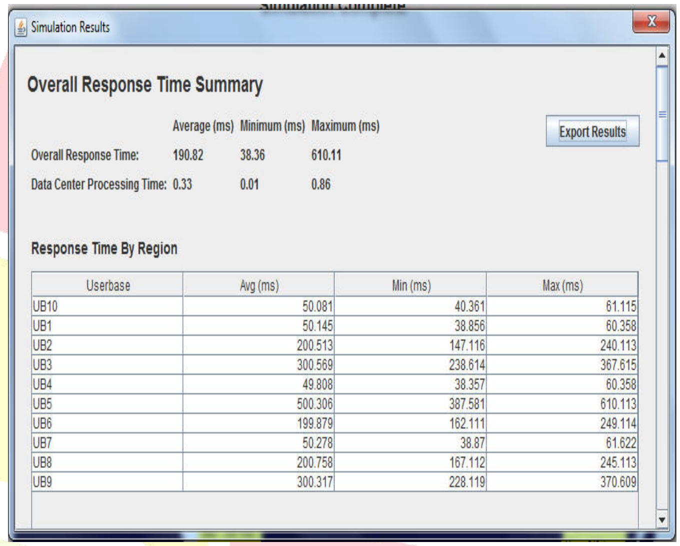
Userbase	Avg (ms)	Min (ms)	Max (ms)
UB10	498.713	410.111	615.11
UB1	50.262	40.609	59.855
UB2	200.172	158.12	252.118
UB3	300.019	238.62	361.613
UB4	499.967	377.631	595.109
UB5	499.363	390.116	615.114
UB6	199.561	165.121	245.114
UB7	50.051	39.86	60.109
UB8	199.167	159.113	245.113
UB9	299.258	231.113	369.117

Figure 8.2 Overall Response Time Summary (For Fig 8.1)

Figure 8.2 shows the response time for each user bases as well as the overall response time. Here, the response time is taken as the measurement for our algorithm.

Figure 8.3 Simulation with 2 Data Centers & 10 User Bases

Figure 8.3 shows the simulation result with two data center and 10 user bases. Here, replication concept takes place. Because, same resource can hold by two different data centers.


Simulation Results

Overall Response Time Summary

	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	190.82	38.36	610.11
Data Center Processing Time:	0.33	0.01	0.86

Export Results

Response Time By Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB10	50.081	40.361	61.115
UB1	50.145	38.856	60.358
UB2	200.513	147.116	240.113
UB3	300.569	238.614	367.615
UB4	49.808	38.357	60.358
UB5	500.306	387.581	610.113
UB6	199.879	162.111	249.114
UB7	50.278	38.87	61.622
UB8	200.758	167.112	245.113
UB9	300.317	228.119	370.609

Figure 8.4 Overall Response Time Summary (For Fig 8.3)

Figure 8.4 shows the response time for each user bases as well as the overall response time. Here, the response time is taken as the measurement for our algorithm.

When compared with single data center concept (Figure 8.2), the overall response time and the response time for each individual user bases are comparatively lower while using more than one data center (Figure 8.4).

IX. CONCLUSION

An efficient replica creation strategy was proposed to improve the data availability in cloud data centers. The authorized user or cloud service provider configures the data center in order to provide appropriate environment to customer. In replica creation module, the copies of data taken as per the frequent usage. The algorithm was proposed for an efficient replica placement. By the result, the response time of each data center was decreased.

X. FUTURE WORK

Future work will be concentrated on the efficient replica handling method in order to reduce the storage space consumption as well as the reduction of cost.

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

REFERENCES

[1] Bakhta Meroufel, Ghalem Belalem (2013) "Managing Data Replication and Placement Based on Availability" AASRI Procedia, Vol.5, pp.147 – 155.

[2] Gang Chen, H.V. Jagadish, Dawei Jiang, David Maier, Beng Chin Ooi (2014) "Federation in Cloud Data Management: Challenges and Opportunities" IEEE Ttransactions on knowledge and data engineering, Vol.26, No.7, pp.1670-1679.

[3] Jan-Jan Wu, Yi-Fang Lin, and Pangfeng Liu (2008) " Optimal replica placement in hierarchical Data Grids with locality assurance" J. Parallel Distrib. Comput. No.68, pp. 1517-1538

[4] Junaid Shuja, Kashif Bilal, Sajjad A. Madani, Pavan Balaji, and Samee U. Khan (2015) "Survey of Techniques and Architectures for Designing Energy-Efficient Data Centers" IEEE Journal, Vol.6, pp.1231-1245.

[5] Mahmoud Al-Ayyoub, Mohammad Wardat, Yaser Jararweh (2015) "Optimizing expansion strategies for ultra scale cloud computing data centers" Simulation Modeling Practice and Theory, Vol.12, pp.1569-1584.

[6] Mr. Sanjay Bansal , Prof. Sanjeev Sharma and Ishita Trivedi "A Dynamic Replica Placement Algorithm to Enhance Multiple Failures Capability in Distributed System" International Journal of Distributed and Parallel Systems (IJDPS) Vol.2, No.6, November 2011.

[7] Mrudula Varade, Vimla Jethani (2013) "Distributed Metadata Management Scheme in HDFS" International Journal of Scientific and Research Publications, Vol.3, Issue.5, 1 ISSN, pp.2250-3153.

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016