ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

Volume 3, Special Issue 3, March 2014

2014 International Conference on Innovations in Engineering and Technology (ICIET'14)

On 21st& 22nd March Organized by

K.L.N. College of Engineering, Madurai, Tamil Nadu, India

An Improved Dynamic Data Replica Selection and Placement in Hybrid Cloud

A.Rajalakshmi, D.Vijayakumar, Dr. K .G. Srinivasagan

PG Scholar, Dept of Computer Science and Engineering, National Engineering College,
Kovilpatti, Tamilnadu, India.

Assistant Professor, Dept of Computer Science and Engineering, National Engineering College,
Kovilpatti, Tamilnadu, India.

Professor & Head Dept of Computer Science and Engineering, National Engineering College,
Kovilpatti, Tamilnadu, India.

Abstract-Cloud computing platform is getting more and more attentions as a new trend of data management. Data replication has been widely used to speed up data access in cloud. Replica selection and placement are the major issues in replication. In this paper we propose an approach for dynamic data replication in cloud. A replica management system allows users to create, register and manage replicas and update the replicas if the original datasets are modified. The proposed work concentrates on designing an algorithm for suitable optimal replica selection and placement to increase availability of data in the cloud. Replication aims to increase availability of resources, minimum access cost, shared bandwidth consumption and delay time by replicating data. Our approach based on dynamic replication that adapts replica creation continuously changing network connectivity and users. The proposed systems developed under the Eucalyptus cloud environment. The results of proposed replica selection algorithm achieve better accessibility compared with other methods.

Keywords- Hash key, Replication, Optimal selection, Eucalyptus, Hybrid cloud, Catalog, virtual synchrony. State transition.

I.INTRODUCTION

Cloud computing is accessing computing resources over the internet as a service. The user has no need to install, no infrastructure to support, just a browser and an internet connection for accessing cloud. Cloud computing means is shared pools of configurable computing Replication used to improves availability by allowing access to the data even when some of the replicas are unavailable. Users access nearby replicas and through increased throughput, In case of failure to redirect the replica.In currently developed cloud systems, static replication is used to copy data to other datacenter where it is most popular data and user request data. Static replication is location unchangeable**.** There is no support for automatic replica creation and data placement. The type of replica provides all user have to access to the required data in on demand manner, enables scalability across multiple locations, and also optimize the use and consumption of network resources. Dynamic replication presents a good approach as decisions are made based on the current access patterns and availability of resources

To address these issues, in replication we need to ensure efficient access and distribution of data based on demand of user. We model the data replica selection problem as a multi objective optimization problem using an optimal replica selection and placement based on access time and response time of transferring data servers to service end point. The method consists of two main phases file application and replication operation. The first phase contains the replica location and creation by using catalog and index. In second phase is used to perform optimization among replicated copies

II. RELATED WORK

Cloud computing one of the major challenges is Data Management and data processing. For data management on the technique used is data replication. The data replication has static and dynamic replication and Synchronous and asynchronous replication strategy, casual consistency model, and on line optimizer algorithm.

Alain Roy et al, (2012) proposed -an elastic replication management system (ERMS) for HDFS. It utilizes a complex event processing engine to distinguish the real time data types. [1]ERMS uses Condor to increase the replication number for hot data in stand by nodes, and to remove the extra replicas after the data cooling down. The erasure codes used to save storage space and network bandwidth when the data becomes cold. ERMS dynamically adapt to changes in data access patterns and data popularity, and impose a low network overhead.

Bakhta meroufel et al (2012) proposed an approach for dynamic replication in a hierarchical grid that takes into account crash failures in the system. Dynamic replication is based on two parameters of data management is availability and popularity [2]. The admin must specify a certain percentage of availability for the data in the system and the no of replicas that satisfies this percentage will be considered as the minimum degree of replication. 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.

Qing song et al (2010) proposed a cost effective dynamic replication (CDRM) scheme for large scale cloud storage system [3]. It builds up a cost model to capture the relationship between availability and replication factor. Replication factor to denote minimum replica number in the metadata. The client contacts the name node with availability setting and block number. Based on this model, lower bound on replica reference number to satisfy availability requirement can be determined. CDRM further places replicas among distributed nodes to avoid the blocking probability, So as to improve load balance and overall performance.

Zeinab Ghilavizadeh (2013) proposed a method of dynamic optimal data replication in data grid it reduces the total job execution time and increase the locality in accessibilities by detecting and impacting factors influencing the data replication. The objective of this paper is to reduce the data access time, to reduce the access latency, to reduce the job execution time to reduce the occupied bandwidth. The proposed method of FORM[4], It consists of two main phases. The second phase is the replacement phase. It introduced a method reduces the total job execution time by grid sites by involving the replication factor. To increase the locality in accessibilities and to replicate data on the grid optimally.

Z.Wang et al (2012) proposed an algorithm for dynamic data replication strategy .It employs historical access records which are useful for picking up a file to replicate. Next one is proactive deletion method, is applied to control the replica number to reach an optimal balance between the read and the write update overhead [9]. This replication algorithm uses the popularity of the file to make replication decision. If the values exceed a threshold, the algorithm decides to do a replication. Different weights to the access record in different time periods for the purpose of finding a popular file. The replication strategy gives up passive deletion method such deleting a replica file when servers do not have enough space to store it.

M. Lei et al (2008) proposed the system-wide data availability problem assuming limited replica storage. Two new metrics to evaluate the reliability of the system proposed an on-line optimizer algorithm [7] that can minimize the Data Missing Rate (MinDmr) in order to maximize the data availability. We then model the problem in terms of an optimal solution in a static system. For online processing of file replication, proposed a novel heuristic algorithm that maximizes the data availability by minimizing the Data Missing Rate (MinDmr) for limited storage resources without sacrificing the data access latency Based on MinDmr,

Ruay-Shiung Chang·Hui-Ping Chang (2008) proposed an algorithm for dynamic data replication mechanism called Latest Access Largest Weight (LALW) is proposed [6]. LALW technique is used to selects popular files for replication and calculates a suitable number of copies for replication. By associating a different weight to each historical data access record, the importance of each record is differentiated depend on access history. The dynamic replication algorithm collects the data access history, which contains file name, the number of requests for file, and the sources that each request came from the user. Historical tables are given different weights according to access. According to access frequencies for all files that have been requested, a popular file is found and replicated to suitable sites to achieve a system load balance.

Houda Lamehamedi and Boleslaw K. Szymanski (2007) proposed replication management algorithm intelligently and transparently places data in strategic locations improving the overall data access performance [8]. Dynamic techniques that adapt replica creation to continuously changing network connectivity and users behavior. A stand-alone framework that is used to provide client access to data. To creates a distributed and decentralized mechanism to replicate and manage access to data that is based on continuous and dynamic evaluation of resource utilization and access performance. The middleware enables each node to monitor and control its local storage space and capacity, access to locally store less, use of network resources, as well as any other available local resources. That is commonly used and popular in data sharing environments and data intensive applications.

Based on the literature survey replication made based on access history, read/write access time and number of replica selection .The main objective our project, during demand of user whether during disaster recovery time we need to concentrate on optimum replication a big challenge in cloud such as replica selection and placement. Major work concentrates on replica selection and placement among hybrid cloud need to perform optimization.

III. PROPOSED SYSTEM

Our proposed system provides incremental scalability and robustness to dynamic data in cloud. Data is replicated; copies of data files are created at many different sites in the cloud. In hybrid cloud, hundreds of nodes involved, and any node failure or network outage can cause data unavailability. Dynamic data replication is used to improve data access in cloud but also reduces the access latency. The proposed work concentrates on optimal replica selection and placement algorithm to increase availability of data in the cloud. Optimizer which includes the replication algorithm that is responsible for automatic replica creation and deletion of replicas. During selection as well as placement of replica needs to consider minimum access cost, bandwidth consumption and delay time.

Architecture Design for Dynamic Data Replication

Fig. 1. Describes dynamic data replication of the system to determine the replica selection and placement. The following modules described about the dynamic data replica selection and placement details

- **Replica Manager**
- **Replica Selection**
- **Index**
- **Replica Catalog**
- **Cloud setup**

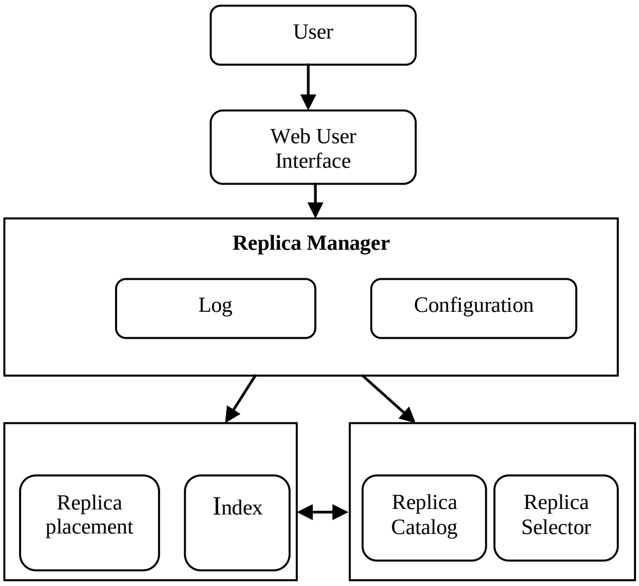

Figure. 1 Architecture of the proposed system

A. Replica Manager

The replica manager directs the creation and management of replicas according to the demands of the users and the availability of storage, a directory keeps track of the all the replicas and their locations. It holds the general information about the datacenter and replica location in the region. It consists of replica services and replica access, Consistency, core replica creation, deletion, and security and authentication.

Replica manager support the management and transferring of data between cloud nodes and creation of new replicas. It also tracks user access patterns, monitor data popularity and also decide need availability if local creation is needed or not. The manager and the catalog are merged into one entity.

B. Replica Management System

- To create replicas on its own by *replication Techniques* that take into account current and future demand for the datasets, locality of requests and storage capacity of the datacenters.
- The *Path manager* **-**route access outgoing request message, handling incoming messages, manage data transfers as well as monitors a node’s connectivity to its neighbors. This also maintains a list of neighbors to which the local node is connected
- *Replica Access and Allocation* replica access module consists of location of data, direction of data flow and user access patterns. It also route outgoing request. It allocates space for newly created replicas and reallocate space from the least recently access.

C. Replica Catalog

- Each newly created file is registered in the replica catalog table. This catalog is also responsible for locating requested data. It maintains the no of user bases, datacenter, replicas in region, no of request at certain time period and availability when each time site store a new replica it send a file register request to replica catalog and then replica catalog add this site to the list of sites and hold the replica. The catalog can be queried by applications to discover the number and the locations of available replicas of a particular dataset.
- A *replica catalog* contains information about locations of datasets and associated replicas and the metadata associated with these datasets. Users query the catalog using attributes to make operations such as locating the nearest replica of a particular database.
- The data search and the location process starts at the local data catalog to check if the data stored and available locally. After receiving a list of possible locations, the local data management service uses network performance tools to choose the source that would yield the best data transfer performance.
- The *Replica Location* replica location initiated to access to a set is needed and the proper request is issued. The request starts the search process that reaches all possible nodes that may have copy of this data set. When multiple locations are discovered, all are reported back to the requester who chooses the most appropriate source node. In dynamic replication platform new node might join the cloud and some nodes might leave depend on demand of user. This creates the need for an adaptive, dynamic approach to discover, locate and access data.

D. Replica Selector

Replica selection requires information about the capabilities and performance characteristics of a storage system. Is based on user demand and failure occur during access time.

An optimization approach for selecting the replica of a data set and a caching function to make informed decisions about local data caching. The ratio of transferred files to the requested file. The ratio of replicated files to the local access. The ratio is smaller means by using optimal replica selection and placement is used.

IV. EXPERIMENT SETUP

In order to do these projects reduce our hardware infrastructure expenditure by using Eucalyptus private cloud, In Figure.2 Shows that for single private cloud set up we installed one front end and three node controllers further we going to add another private cloud for performing hybrid cloud.

In the Frontend and Node Controller configuration, the Cloud controller, Walrus, Cluster controller, and Storage controller are installed on one machine, called the Frontend. The Node controller is installed on other machines. In this configuration we can have one Frontend and one or more Node controller.

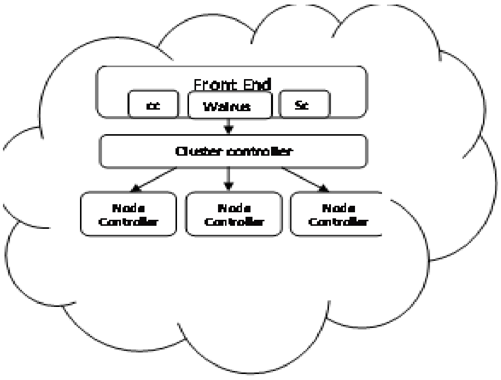

Figure. 2.Eucalyptus private cloud setup

The dynamic changes of data replicas and network conditions in private cloud computing, as well as the constraints of different users and the flexibility of selection criteria, the dynamic data replica and selection by using optimization to solve the data replica selection problem.

In our approach, the work has been done in application level with the help of dot net framework.Figure.3. shows in client interface requesting is to upload file in cloud datacenter meanwhile receive the file information domain address and bandwidth utilization by using file transfer protocol. Getting file details such as file, size of the file, CPU and memory usage utilization are received from the cloud storage. Our proposed dynamic replica selection and placement algorithm is using bandwidths among the datacenters are calculated based on where the user Read/Write and upload operations are performed. During write operation major focus optimization among the original and replicated copies. Based on the bandwidth utilization the file needs to replicate or not, the no of copies make decision based on same file uploaded. If less bandwidth is consumed when data is dynamically replicated based on user demands. If user will different file name has been provide means the content is same is search by using hash key value. We concentrate on location of replicated copies. After verifying DHT the data is decided to take up a replication operation. In Datacenter designed, bind Isis2 for making consistency among replicas. If same file make replication isis2 support the Key value stores, it avoids the more unwanted copies by using DHT (Distributed Hash Table).

A.Isis2 Library

Isis2 is a new open platform for data replication in the cloud C# Library (callable from any .NET language) offering replication techniques for cloud computing developers. Based on this model fuses virtual synchrony and state machine replication models. Isis is a prebuilt technology that automates many of the hard tasks involved in replicating services and the data on which they depend. Isis2 is a very simple library interface that hides as much of this complexity as feasible.

To run an application you have compiled with Mono would be to invoke it through the Mono runtime we integrate the two systems by using Linux in Eucalyptus front end support. The application is deployed in Eucalyptus private cloud. User can bundle her own root file system, and upload and then register this image link it with a particular kernel and ram disk image. This image is uploaded in to a user –defined bucket within walrus, and can be retrieved anytime from any availability zone. This allows users to create specialty virtual appliances and deploy them within eucalyptus with ease.

A library to create highly resilient, secure, scalable applications. The methods automate important tasks. It maintains

- Consistent replicated data
- Coordinator/Cohort fault-tolerance
- Synchronization/Multicast operations
- Locking(read/write)
- Shading/ordering
- Key-value Stores
- Failure sensing

B. Analysis

Response time is based on bandwidth utilization of each random file selection. The utility computing based on scaling and weighting depending on the user the replication selection and placement is performed.

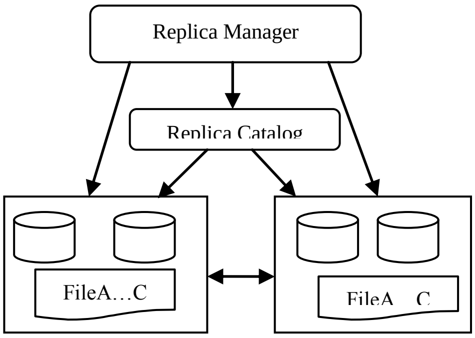

Figure. 3. Replica Management using FTP

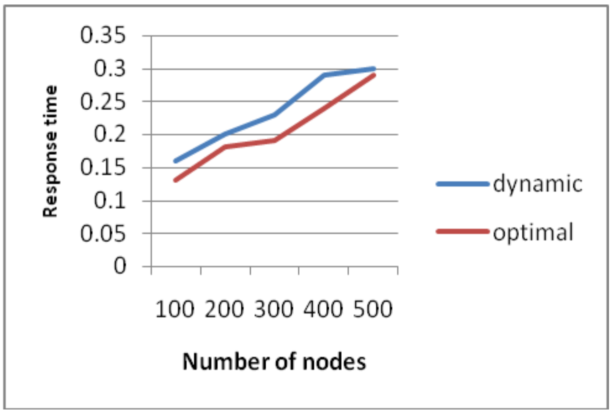

Figure. 4. Number of nodes in dynamic and optimal approaches

In the next step of experimental results the no of user access and failure are calculated. Figure. 4. Describes Results are collected for different scenarios where both storage availability at each node and the replication threshold are updated.

The no of replica utilization up to 80% because of the storage consideration in future demand of user .In each file the no of replica are increase/decrease based on the optimum selection the replication has been made. In any other replica the life time of data whether no of replica, no need to go for separate replica. We set a own private cloud, in any user from outside can utilize the cloud we need to consider the replication cost

$$R_{cost} = \frac{U_{pd} + T_c + RW_c}{No.of\ Replica}$$

R_c=Replicacost,U_{pd}=Unitperdata,T_c=Transfer cost,RW_c=Read/Write operation

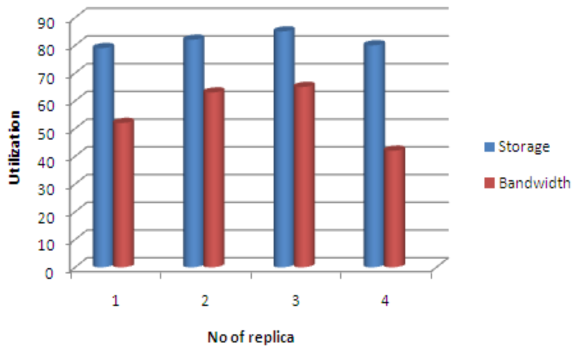

Figure5.Calculate utilization among storage, bandwidth,

In Figure.5. Shows that utilization among storage, bandwith, cost. Bandwidth utilization is based on storage, size of data, data change rate, link speed.

V.CONCLUSION

In this paper we proposed a dynamic replica selection and placement architecture for managing replicas in cloud environment. Our experimental result shows that an optimal replica selection algorithm transparently places data in strategic locations an improved the overall data access performance and bandwidth utilization. In our future work to extend our approach replica placement in geographical locations of our further enhancement.

REFERENCES

[1]. Zhendong Cheng, Zhongzhi Luan, Alain roy, Ning Zhang and Gang Guan (2012) 'ERMS:An Elastic Replication Management System for HDFS', IEEE International Conference cluster computing, pp .32-40.

[2]Bakhta Meroufel and Ghalem Belalem (2012) 'Dynamic Replication Based on Availability and Popularity in the Presence of Failures, Journal of Information Processing Systems', Vol.8, No.2, pp. 263-278.

[3].Wei Q, Veeravalli B, Gong B, Zeng L, Feng D. CDRM: Acost-effective dynamic replication management scheme for cloud storage cluster. In Proc. 2010 IEEE International Conference on Cluster Computing, Heraklion, Crete, Greece, Sept. 20-24, 2010, pp.188-196.

[4]. Ghilavizadeh Zeinab,seyed javad mirabedini,Ali Harounabadi (2013), 'A new Fuzzy optimal data Replication for data Grid', Management science,pp.927-936.

[5] Rafah m.Almuttairi et al.,'Two phased service oriented broker for replica selection in data grids',2012,pp.953-972

[6] Ruay-shiung chang ,'A dynamic data replication strategy using access weights in data grids,2008,spiringer,pp.278-294

[7] Lei M, Vrbsky S V, Hong X. 'An on-line replication strategy to increase availability in data grids'. Future Generation Com-puter Systems, 2008, 24(2) . pp 85-98.

[8] Lamehamedi.H and Szymanski.B (2007) 'Decentralized data management framework for data grids' Future Generation computer system, pp.109-115

[9] B.-y.Ooi et al,'Dynamic service placement and replication framework to enhance service availability using team formation algorithm', Elsevier Journal of System and Software, March 2012, pp.2048-2062.

[10] Da-Wei Sun, Gui -Ran Chang et.al,Journal of Computer Science and Tech,2012, Vol.1.'Modeling a Dynamic Data Replication Strategy to Increase System Availability in Cloud Computing Environment'.pp256-272

[11]Tung nguyen et al.,"Differentiated Replication strategy in data centers, IFIP international federation of information processing 2010, pp.276-287.

[12]http://www.tutorialspoint.com/asp.net/asp.net_file_uploading.html

[13] http://www.asp.net/mvc/tutorials

[14]http://msdn.microsoft.com/enus/library/bb386403%28VS.100aspx.

[15] Isis2.codeplex.com

[16] Tung Nguyen (2010) 'Differentiated Replication Strategy in Data Centers', IFIP-International Federation for Information Processing, pp.277-288

[17] http://www.w3schools.com/aspnet/aspnet_controls.asp

[18] Wang. Z, Xiong .N and Pan.Y (2011) 'A novel dynamic network data replication scheme based on historical access record and proactive deletion', Springer, pp 288.

[19]http://aws.amazon.com/articles/1904/