Hosting a Multi Cloud data in a Cost Effective way to achieve High Availability

Yashaswini H S[1], Manjunath k. G[2]

[1]M.Tech Student, Dept. of Computer Science & Engineering, SIT, Tumakuru, Karnataka, India.
[2]Assistant Professor, Dept. of Computer Science & Engineering, SIT, Tumakuru, Karnataka, India.

---***---

Abstract - *In the continually expanding worldwide present day drifts, the worldwide arrangement of cloud datacenters is empowering enhanced execution and conveyance of quick reactions. Cloud computing is another and quickly developing innovation that offers an imaginative, proficient and versatile plan of action for associations. Cloud computing is sort of processing where assets are shared instead of having neighborhood servers or individual gadgets to handle different applications. More undertakings and associations are facilitating their information into the cloud, to diminish the IT upkeep cost and improve the unwavering quality of information. Existing arrangements are restricted to cloud-if capacity, which offers low execution taking into account inflexible cost plans. Here, we propose a cost proficient multi-cloud for the data with high accessibility. The proposed plan contains two principle works; the first is selecting a few reasonable mists and a fitting repetition procedure to store information with minimized monetary cost and ensured accessibility. The second is setting off a move procedure to alarm the appropriation of information as per the varieties of information access example and valuing of mists. The proposed technique not just spares around 20 percent of money related cost additionally shows sound flexibility to information and value changes.*

Key Words: **Multi-cloud, data hosting, cloud storage**

1. INTRODUCTION

Cloud computing is a trend in the present day scenario with almost all the organizations are entering into it. Cloud computing is the collection of virtualized and scalable resources and provide service based on "pay only for use" strategy. It is a faster profit growth point by providing a comparably low-cost, scalable, position-independent platform for clients" data. It is constructed based on open architectures and has the capability to incorporate multiple internal and/or external cloud services together to provide high interoperability. Such a distributed cloud environment is called **Multi-Cloud.**

Recent years have witnessed a "gold rush" of online data hosting services (or says cloud storage services) such as Amazon S3, Windows Azure, Google Cloud Storage, Aliyun OSS [1], and so forth. These services provide customers with reliable, scalable, and low-cost data hosting functionality.

1.1 Multi-cloud data hosting

More and more enterprises and organizations are hosting all or part of their data into the cloud, in order to reduce the IT maintenance cost (including the hardware, software, and operational cost) and enhance the data reliability [2], [3], [4]. Recently, multi-cloud data hosting has received wide attention from researchers, customers, and startups.

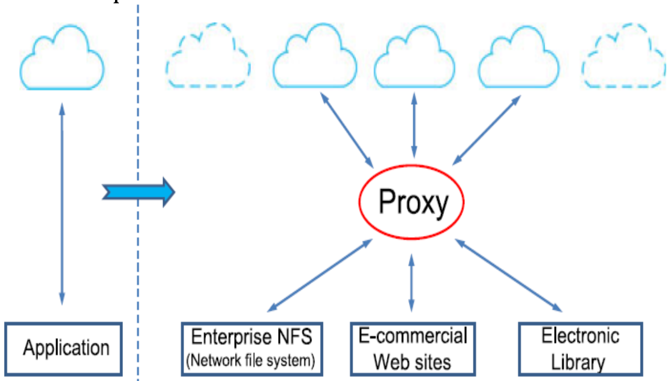

Fig-1: Basic principle of multi-cloud data hosting.

The basic principle of multi-cloud (data hosting) is to distribute data across multiple clouds to gain enhanced redundancy and prevent the vendor lock-in risk, as shown in Fig. 1. The "proxy" component plays a key role by redirecting requests from client applications and coordinating data distribution among multiple clouds.

2. BACKGROUND

The cloud computing environment comprises of five characteristics, three delivery models and four deployment models. The five important characteristics of cloud computing are comprising first stratum are: location-independent resource pooling that is provider resources pooled to server multiple clients, on-demand self-service, rapid elasticity which is ability to quickly scale in/out service, broad network access, and measured service that is renting the services use per pay basis.

Three Cloud Delivery models are Iaas, PaaS and SaaS comprises middle stratum of cloud computing environment.

In Software as a Service (SaaS), applications are there that are enabled for the cloud. It supports an architecture that can run multiple instances of it-self which are location independent. This is nothing but a monthly subscription based pricing model and it is stateless. Examples of SaaS are MobileMe, Google docs, Zoho.

Platform as a Service(PaaS) includes platform on which developers can write their applications to be run on cloud environment. This platform normally has multiple application services available for quick deployment. Examples of PaaS are Google App Engine, Microsoft AZURE, Force.com.

Infrastructure as a Service (IaaS) used by consumer by providing storage, processing, networking, and other fundamental computing resources where the consumer is able to deploy and run software, which can include operating systems and applications. It is highly scaled redundant and shared computing Infrastructure approachable using internet technologies. Examples of this type of delivery model include Amazon EC2, Sun's cloud services, Terremark cloud offering etc.

Third stratum in the cloud computing environment consists of cloud deployment models which include public, private, community, and hybrid clouds. A cloud architecture which can be accessed by multi-tenants and is available to the public is called a public cloud. Cloud which is available for a particular group is private cloud , while a community cloud is modified for a specific group of consumers. Hybrid cloud infrastructure is a combination of two or more clouds [5].

3. EXISTING METHODS

Z. Li, C. Jin, T. Xu, C. Wilson, Y. Liu, L. Cheng, Y. Liu, Y. Dai,and Z.-L. Zhang [3], in this paper the author introduced the various advantages of Cloud storage services like Google drive, Drop box and many other. In the existing industrial and enterprise based data hosting systems, data availability are usually guaranteed by replication or erasure coding. For replication, replicas are put into several clouds, and a read access is only served by the cheapest cloud, such that the cloud that charges a minimal amount for out-going bandwidth and GETS operation.

Data replication is generally adopted for systems with Distributed Applications that are hosted in a Hybrid Cloud and often require access to the same data from various parallel application components. If the application components accessing the data are globally distributed, then the data access performance for the parallel systems may be reduced drastically, comparing if data is only stored in one geographic location. Some of these locations may only handle a subset of the available data or the data has to be obfuscated, based on the set of governing and operating laws and corporate regulations.

The other method is the erasure coding. An erasure code is a forward error correction (FEC) code for the binary erasure channel, in which a message of k symbols is transformed into a longer message (code word) with n symbols, such that the original message can be recovered only from a subset of the n symbols. Here, the fraction r = k/n is called the code rate, the fraction k‟/k, is called the reception efficiency, where k‟ denotes the number of symbols required for recovery. In cloud computing, data is encoded into „n‟ blocks including „m‟ data blocks and „n≠ m‟ coding blocks, and these blocks are put into „n‟ different clouds. In this case, the data availability can be guaranteed within a lower storage space, but a read access has to be served by multiple clouds that store the corresponding data blocks.

In the multi-cloud scenario also, the replication techniques and the erasure coding mechanisms are used to meet different availability requirements but the implementation of these are very different.

3.1 Disadvantages of the Existing System

- Security and availability of the system, user data vulnerable to potential attacks.
- A quarantined and secure hybrid architecture ensuring a dependable and reliable service is missing.

4. PROPOSED METHOD

In this paper, we propose a novel cost-efficient data hosting scheme with high availability in heterogeneous multi-cloud, named "CHARM".

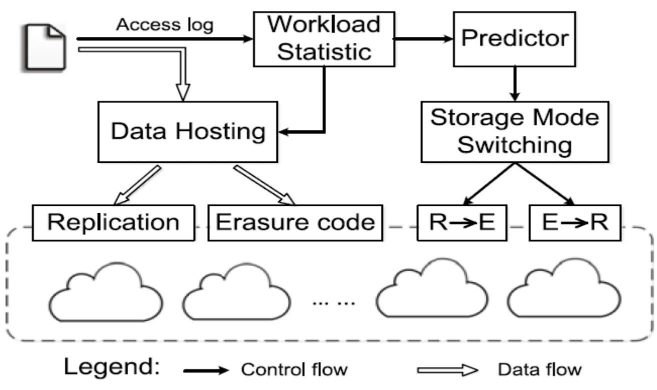

Fig-2: The architecture of CHARM. "R" represents replication and "E" represents erasure coding.

It intelligently puts data into multiple clouds with minimized monetary cost and guaranteed availability. Specifically, we combine the two widely used redundancy mechanisms, i.e., replication and erasure coding, into a uniform model to meet the required availability in the presence of different data access patterns.

Next, we design an efficient heuristic-based algorithm to choose proper data storage modes (involving both clouds and redundancy mechanisms). Moreover, we implement the necessary procedure for storage mode transition (for efficiently re-distributing data) by monitoring the variations of data access patterns and pricing policies.

4.1 Heuristic Based Algorithm

Algorithm:

Input: file size S, read frequency c_r, n's upper limit ξ
Output: minimal cost C_{sm}, the set ψ of the selected clouds
$C_{sm} \leftarrow$ inf; $\psi \leftarrow \{\}$
$L_s \leftarrow$ sort clouds by normalized $\alpha a_i + \frac{\beta}{P_i}$ from high to slow
for $n = 2$ to ξ **do**
  $G_s \leftarrow$ the first n clouds of L_s
  $G_c \leftarrow L_s - G_s$
  for $m = 1$ to n **do**
    $A_{cur} \leftarrow$ calculate the availability of G_s
    if $A_{cur} \geq A$ **then**
      $C_{cur} \leftarrow$ calculate the minimal cost
      if $C_{cur} < C_{sm}$ **then**
        $C_{sm} \leftarrow C_{cur}$
        $\psi \leftarrow G_s$
      end
    else
      /*heuristically search better solution*/
      $G_s \leftarrow$ sort G_s by a_i from low to high
      $G_c \leftarrow$ sort G_c by P_i from low to high
      for $i = 1$ to n **do**
        flag $\leftarrow 0$
        for $j = 1$ to $N - n$ **do**
          if $a_{G_c[j]} > a_{G_s[i]}$ **then**
            swap $G_s[i]$ and $G_c[j]$
            flag $\leftarrow 1$
            break
          end
        end
        if *flag* = *0* **then**
          break
        end
        $A_{cur} \leftarrow$ calculate the availability of G_s
        if $A_{cur} \geq A$ **then**
          $C_{cur} \leftarrow$ calculate the minimal cost
          if $C_{cur} < C_{sm}$ **then**
            $C_{sm} \leftarrow C_{cur}$
            $\psi \leftarrow G_s$
          end
          break
        end
      end
    end
  end
end
return C_{sm}, ψ

Algorithm 1: Heuristic Based

The key idea of this heuristic algorithm can be described as follows: We first assign each cloud a value which is calculated based on four factors (i.e., availability, storage, bandwidth, and operation prices) to indicate the preference of a cloud. We choose the most preferred n clouds, and then heuristically exchange the cloud in the preferred set with the cloud in the complementary set to search better solution. This is similar to the idea of Kernighan-Lin heuristic algorithm, which is applied to effectively partition graphs to minimize the sum of the costs on all edges cut. The preference of a cloud is impacted by the four factors, and they have different weights. The availability is the higher the better, and the price is the lower the better.

4.2 Storage Mode Transition Algorithm

Algorithm:

Input: the generated table Γ, the ith file's current storage mode $M[i]$, current read frequency $R[i]$, file size $S[i]$
Output: void
$dSize \leftarrow$ the size dimension of Γ
$dRead \leftarrow$ the read frequency dimension of Γ
for *each file i* **do**
  for j *in* $len(dSize)$ **do**
    if $S[i] \geq dSize[j]$ **then**
      $dS \leftarrow j$
    else
      break
    end
  end
  for j *in* $len(dRead)$ **do**
    if $R[i] \geq dRead[j]$ **then**
      $dR \leftarrow j$
    else
      break
    end
  end
  if $M[i] \neq \Gamma[dS][dR]$ **then**
    $T \leftarrow$ monetary cost of transiting from $M[i]$
    if $M[i] > \Gamma[dS][dR] + T$ **then**
      transit from $M[i]$ to $\Gamma[dS][dR]$
    end
  end
end

Algorithm 2: Storage Mode Transition

The storage mode table can be calculated in advance, because it is only affected by the available clouds, their pricing policies, and availabilities. When deciding the storage mode for each file, we use the read frequency and the size of the file to look up the table for the corresponding storage mode. We calculate the storage mode for each file using its predicted read frequency in the time interval t. If the storage mode is different from the previous one, we change the storage mode of this file. The storage mode transition does not impact the performance of the service. Since it is not a latency-sensitive process, we can decrease the priority of transition operations, and implement the transition in batch when the proxy has low workload.

4.3Advantages

- A multi-cloud approach is where an enterprise uses two or more cloud services, hence minimizing the risk of widespread data loss or outage due to a component failure, as in a single cloud computing environment.
- Data Hosting with low cost cloud storage Server will make effective use for client benefit.

5. EXPERIMENTAL RESULTS

We conduct extensive simulations to evaluate the performance of our scheme. The simulations are driven by two typical real-world traces. We first briefly introduce the two collected traces and present the evaluating methodology, then show the performance of our scheme. At last, to make the results more convincing, we also implement the prototype experiments on top of four mainstream commercial clouds, the results of which prove the correctness of the simulations and the efficiency of CHARM. We implemented the prototype experiments on four mainstream commercial clouds: Amazon S3, Windows Azure, Google Cloud Storage and Drop Box, and pick 10 different data centers2 from them. We created accounts in the four clouds, and replayed Amazing Store trace and Corsair trace for a whole month, using three different schemes (i.e., CHARM, RepGr, EraGr) which choose their preferred data centers from the 10 available ones.

Items	Amazingstore					
	RepGr		EraGr		CHARM	
	P	S	P	S	P	S
Bandwidth	4.4778	4.4805	7.2530	7.2982	4.5413	4.6216
Storage	3.4102	3.4198	1.2945	1.2982	1.6356	1.6394
Operation	0.0004	0.0001	0.0130	0.0059	0.0010	0.0005
Total	7.8884	7.9004	8.5605	8.6023	6.1779	6.2614

The detailed results are shown in the above Table 1. Here P represents for prototype experiment and S represents for Simulation experiments. For the prototype experiments, CHARM outperforms RepGr and EraGr by 21.7 and 27.8 percent for Amazing Store trace, and similarly the savings are 37.8 and 13.8 percent for Corsair trace, which proves the efficacy of CHARM.

6. CONCLUSION AND FUTURE WORK

Cloud services are experiencing rapid development and the services based on multi-cloud also become prevailing. One of the most concerns, when moving services into clouds, is capital expenditure. So, in this paper, we design a novel storage scheme CHARM, which guides customers to distribute data among clouds cost-effectively. CHARM makes fine-grained decisions about which storage mode to use and which clouds to place data in. The evaluation proves the efficiency of CHARM. This system can be enhanced by developing an automatic update methodology that updates the required fragments only. In future this system will save the time for downloading and uploading the file again.

REFERENCES

[1] Aliyun OSS (Open Storage Service). [Online]. Available: http:// www.aliyun.com/product/oss, 2014.
[2]Gartner: Top 10 cloud storage providers.[Online]. http://www.networkworld.com/news/2013/010313 gartnercloud-storage-265459.html?page=1, 2013.
[3] Z. Li, C. Jin, T. Xu, C. Wilson, Y. Liu, L. Cheng, Y. Liu, Y. Dai, and Z.-L. Zhang, "Towards network-level efficiency for cloud storage services," in Proc. ACM SIGCOMM Internet Meas. Conf., 2014, pp. 115–128.
[4] Z. Li, C. Wilson, Z. Jiang, Y. Liu, B. Y. Zhao, C. Jin, Z.-L. Zhang, and Y. Dai, "Efficient batched synchronization in dropbox like cloud storage services," in Proc.ACM/IFIP/USENIX 14th Int. Middleware Conf., 2013, pp. 307–327.
[5] Mohammed A. AlZain, Eric Pardede, Ben Soh, James A. Thom, "Cloud Computing Security: From Single to Multi-clouds", hicss, pp.5490-5499, 2012 45th Hawaii International Conference on System Sciences, 2012.