International Journal of Latest Trends in Engineering and Technology
Special Issue NCRTCC 2017, pp.019-025
e-ISSN:2278-621X

TRIONES BASED DATA PLACEMENT OPTIMIZATIONFOR MULTI -CLOUD STORAGE

G.Keerthiga[1]

Abstract—Nowadays, more and more enterprises and organizations are hosting their data into the cloud, in order to reduce the IT maintenance cost and enhance the data reliability. Socially aware services often have a large user base and data of users have to be partitioned and replicated over multiple geographically distributed clouds. Choosing in which cloud to place data however is difficult as well as many organizations migrate their on-premise software systems to the cloud. However, current coarse-grained cloud migration solutions have made a transparent migration of on-premise applications to the cloud a difficult, sometimes trial-and-error based endeavor. Triones applies nonlinear programming to define the problem of data placement optimization under complex requirements. In such cases, erasure coding is usually applied for further improvement as it can significantly reduce the cost of data storage compared to full replication. The total amount of data that must be transferred over the network can also be reduced. With erasure coding, each data object is evenly divided into k blocks and then these blocks are used to generate n, k encoded data blocks (n blocks in total, n > k). This is called as parameter n; k of erasure coding. These n data blocks will be uploaded into n cloud storage providers, with each provider holding just one data block respectively. Any k blocks from the n ones can be used to reconstruct the original data object. We can see that erasure coding can naturally fit into the multi-cloud storage.Scalia has done the optimization work on the cost. However, it only conductedsingle objective optimization, whichis far from enough for the multi-cloud storage. Multi-objective optimizationsuch as optimizing cost, access latency, and fault-tolerance at the same time has been left unaddressed. In fact, the most important part of optimizingthe multi-cloud storage is the optimization on data placement, which is to choose an optimized data placement configuration. A data placement configuration in the multi-cloud storage consists of erasure coding parameter n; k and n cloud storage providers.

Index Terms—data placement optimization,complex requirements,migration,access latency.

I. INTRODUCTION

Cloud providers are offering efficient on-demand storagesolutions that can virtually scale indefinitely. Internetservices that span multiple geographically distributed clouds intrinsically have multiple system objectives, including budgeting the monetary expenditure spent on cloud resource usage ensuring the service quality perceived by users (e.g., access latency) and even reducing the carbon footprint of the service, to name a few. The data placement problem is particularly challenging for multi-cloud services that are socially aware, where users build social relationships and share contents with one another, as reflected by Online Social Network (OSN) services and many non-OSN services with social components. In fact, the most important part of optimizing the multi- cloud storage is the optimization on data placement, which is to choose an optimized data placement configuration. A data placement configuration in the multi-cloud storage consists of erasure coding parameter (n, k) and n cloud storage providers. It is non-trivial to achieve data placement optimization in general in the multi-cloud storage. Data

[1] *New Prince Shri Bhavani College of Engineering and Technology, Gowriwakkam, Chennai – 73*

placement configurations with diverse cloud storage providers and different n, k parameters offer totally different serving ability (e.g., cost, access latency, or vendor lock-in). Moreover, it further complicates the optimization that developers do have different optimization requirements, according to the properties of their systems or applications. Here, we use simple requirement to represent single objective optimization, i.e., optimizing only one factor. Complex requirement is used to represent the situation of optimizing multiple factors at the same time. For instance, some developers who want to store critical data in the multi-cloud storage would be interested in optimizing factors including cost, fault-tolerance level (FTL), and vendor lock-in level simultaneously. In this paper, we present Triones, a systematic model to formulate data placement in the multi-cloud storage as well as the ways of its optimization. Triones applies non- linear programming to define the problem of data placement optimization under complex requirements. In the non-linear programming, Triones regards all the factors associated with the multi-cloud storage as derived variables from basic variables and constants. Basic variables are used as representation of the final optimized results (configurations) for data placement. They consist of a set of 1 or 0 to represent whether a corresponding provider is used or not with an additional variable k, representing the parameter of erasure coding. Besides, constants stand for the characteristics of underlying cloud storage providers. The objective function of the non-linear programming is a combination of factors to be optimized. They map to a complex requirement demanded by system or application developers.

Moreover, inequalities in the non-linear programming are used to represent constraints on factors. The left-hand side of each constraint inequality represents one factor under constraint while the right-hand side of it is the quantifiable constraint on this factor. We use Euclidean distance measure in an abstract geometric space, to balance among different kinds of factors to get the optimized results for the objective function. The results correspond to the optimized data placement configurations that satisfy developer's complex requirements subject to certain constraints. In addition, Triones can also address data placement optimization under simple requirements, in which cases the objective function contains only one factor.

II. RELATED WORK

Many storage systems have been built using public cloud services. Exploited rented virtual machines(VMs), while backup ,file and database systems have been built using public cloud storage services (e.g.,AmazonS3 or Windows Azure Storage). Some systems improve integrity and security via auditing and encryption. However, these systems are within the domain of a single cloud and suffer from reliability and vendor lock-in issues. Many studies focus on security and privacy aspects which are major obstacles in cloud adoption for both individuals and companies.
mLibCloud[1] The schemas or models in their systems only randomly chose data placement configurations to achieve certain features. Compared with them, Triones is a systematic model to address the optimization issue for developers in the multi-clouds storage. It enables them to deploy their systems or applications in an optimized way under their simple or complex requirements. One similar work to Triones is Scalia.
Thanasi's G. Papaioannou [2] Scalia used an adaptive scheme to choose different data placement configurations for offering the optimal cost while satisfying certain constraints. However, from Triones' point of view, the model in Scalia only conducted single objective optimization. Triones does the work for both single objective as well as multi-objective optimization.
Scalia was inspired by RACS[3], which employs RAID at the cloud storage level, making also use of erasure codes instead of full replication[4].
However, RACS does not adapt data placement to different conditions to meet any optimization objectives. K. D. Bowers and A. Juels and A. Opera., HAIL[5], distributes redundant blocks of a file across multiple servers, while allowing a client to make sure that the file is not corrupted even in the case of a server compromise use an hybrid model[6], Peer-to-peer distributed hash tables (DHTs) propose a logically centralized, physically distributed, hash table abstraction that can be shared simultaneously by many applications. Ensuring that data objects in the DHT have high availability levels when the nodes that are storing them are not themselves 100% available requires some form of data redundancy. Peer-to-peer DHTs have proposed two different redundancy schemes: replication and erasure coding.
This paper aims to provide a comprehensive discussion about the advantages of each scheme. While previous comparisons exist they mostly argue that erasure coding is the clear victor, due to huge storage savings for the same availability levels (or conversely, huge availability gains for the same storage levels). Our conclusion is somewhat different: we argue that while gains from coding exist, they are highly dependent on the characteristics of the nodes that comprise the overlay. In fact, the benefits of coding are so limited in some cases that they can easily be outweighed by some disadvantages and the extra complexity of erasure codes. We begin this paper by performing an

analytic comparison of replication and coding that clearly delineates the relative gains from using coding vs. replication as a function of the server availability and the desired DHT object availability. We present a model that allows us to understand server availability. Then we use measured values from three different traces to find out exact values for the parameters of the model. This allows us to draw more precise conclusions about the advantages of using coding or replicationis adaptive to the various pricing and resource conditions, soas to dynamically find the optimal data placement

III.PROPOSED SYSTEM

It has been a trend that large numbers of organizations, companies, government departments, are storing their data into cloud. However, only using one cloud storage provider is More likely to suffer from single-point failures and vendor lock-in. As a result, the multi-cloud storage that relies on multiple cloud storage providers to place data at some redundancy level has been used by current works. It can provide better service quality including vendor lock-in avoiding as well as fault-tolerance improving.

3.1 NON-LINEAR TECHNIQUE

Triones addresses the problem of data placement optimization by applying non-linear programming and geometric space abstraction. It could satisfy complex requirements involving multi-objective optimization. Secondly, Triones can effectively balance among different objectives in optimization and is scalable to incorporate new ones. The effectiveness of the model is proved by extensive experiments on multiple cloud storage providers in the real world.

3.2 PROBLEM STATEMENT

Triones applies nonlinear programming to define the problem of data placement optimization under complex requirements. In the non-linear programming, Triones regards all the factors associated with the multi-cloud storage as derived variables from basic variables and constants.

Basic variables are used as representation of the final optimized results (configurations) for data placement. They consist of a set of 1 or 0 to represent whether a corresponding provider is used or not with an additional variable k, representing the parameter of erasure coding.

IV. SYSTEM ARCHITECTURE

Architecture diagram shows the relationship between different components of system. This diagram is very important to understand the overall concept of system. Architecture diagram is a diagram of a system, in which the principal parts or functions are represented by blocksConnectedby lines that show the relationships of the blocks.

They are heavily used in the engineering world in hardware design, electronic design, software design, and process flow diagrams

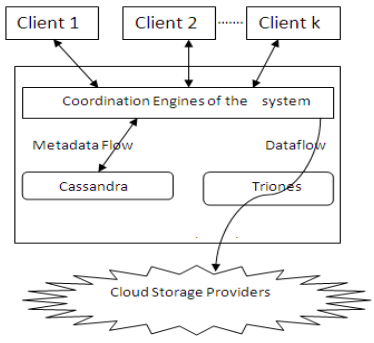

Fig 1.System Architecture

V. REGISTERATION TO CENTRAL AUTHORITY

User first store the pages and access the cloud generally get the permission from the Central authority (CA).
The CA requires the pages to be maintained.It supports the key based data access.

5.1 CONSUMER REQUEST THE ACCESS

Consumers enter the details and store the data in cloud. The query retrieves the data from a cloud that matches the entire accessible performances

VI. MULTI CLOUD STORAGE

Multi-cloud is the use of multiple cloud computing services in a single heterogeneous architecture.For example, an enterprise may concurrently use separate cloud providers for infrastructure (IaaS) and software (SaaS) services, or use multiple infrastructure (IaaS) providers. In the latter case, they may use different infrastructure providers for different workloads, deploy a single workload load balanced across multiple providers (active-active), or deploy a single workload on one provider, with a backup on another (active-passive).
There are a number of reasons for deploying a multi-cloud architecture, including reducing reliance on any single vendor, increasing flexibility through choice, and mitigating against disasters. It is similar to the use of best-of-breed applications from multiple developers on a personal computer, rather than the defaults offered by the operating system vendor. It is a recognition of the fact that no one provider can be everything for everyone. It differs from hybrid cloud in that it refers to multiple cloud services rather than multiple deployment modes (public, private, legacy).

VII. ALGORITHM USED IN TRIONES TRIONES:

A systematic model to formulate data placement in the multi-cloud storage as well as the ways of its optimization. Triones applies nonlinear programming to define the problem of data placement optimization under complex requirements. In the non-linear programming.

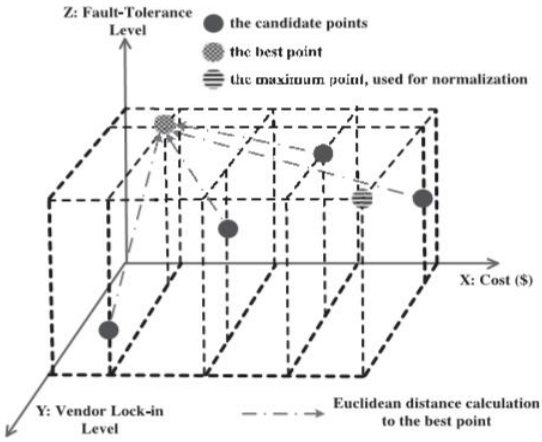

Fig. 2. A three-dimension geometric space for optimizing cost, fault tolerance, and vendor lock-in level.

Triones addresses this issue in the way through multi- dimension geometric space abstraction. In this geo- metric space, each dimension independently represents one factor. Based on the definition of these factors, every possible data placement configuration (i.e., Xs) can be used to calculate a specific value on each factor. Then each configuration maps to a point in this space, with the components of the point being the values of factors
F={ f1,f2,...,fE}.
E is the total number of factors in the objective function. Configurations whose values of factors violate the constraints do not mapto anypointinthe multi dimensiongeometricspace. As illustrated in Fig. 2, for optimization, Triones sets a best point in the multi-dimension geometric space. The best point consists of the best value in each dimension. For example, in the dimension of access latency, the best value should be the lowest access latency to a specific data placement configuration. Notice that not all the best values are the smallest ones for corresponding

dimensions. The best means the smallest for cost and vendor lock-in level while it means the largest for other factors such as availability and fault-tolerance level. It is decided by the definitions of the factors under consideration. Then, each point can get the distance to the best point by using Euclidean distance measure. A point with a shorter distance to the best point will be considered as a better point and the corresponding configuration will be considered as a better schema. However, the Euclidean distance requires the same units of measurement for each dimension. Thus, we have to calculate the distance by using normalized values instead of absolute ones. The normalized value of a component in a point is calculated by its absolute value over maximum value in the corresponding dimension. The maximum value of every dimension composes a maximum point, which is also used for euclidean distance measure. The best point and the maximum point are usually virtual points that donot match any data placement configuration. They are only meaningful for getting the optimized point (configuration). With normalized values in all dimensions, Triones provides a general objective function in the form of euclidean distance measure. Assume that there are E factors to be optimized. Then in a E-dimension geometric space, a data placement configuration Xs could map to a point (f1(xs,P),…. fE(Xs,P)). Moreover, the best point could be labeled as (f1best,… fEbest)and the maximum point could be labeled as((f1max,… fEmax). For the best and maximum points, fEbest (e= 1,...,E) is the best value among fe(Xs,P) (1 ≤ s ≤ S) while femax (e = 1,...,E) is the maximum value among (1 ≤ s ≤ S). Then the euclidean distance from point of Xs to the best point, denoted as EDs, can be defined as:

$$EDs = \sqrt{\sum_{e=1}^{E} W_e \times (f_e(Xs, P) - fbest_e)^2}$$

Femax

We is the optimization weight for factor fe and ∑Ee=1 We= 1. When developers require some factors to have more importance than others in the optimization, they can increase the optimization weight of these factors. To get a final result that optimizes (minimizes) this objective function, Triones traverses all possible data placement configurations. For each configuration, the value of each optimizing factor it could offer will be calculated. Then the configuration maps to a point in the multi-dimension geometric space. By setting a best point and a maximum point from the components of all points, each EDs can be gotten. Thus, the configuration with the smallest value of EDs corresponds to the optimized result of the objective function.

7.1 SOURCE CODE

Algorithm 1. Optimization Results Calculation
Input: Rul:cfg for the Rules:cfg, Sys:cfg for the System
Patterns:cfg, Pro:cfg for the Providers:cfg, lastTime,
success
Output: Opt:cfg for the OptimizationResults:cfg
1: global P½_½_ “Constant coefficient matrix
2: providers getProviders Pro:cfgÞ
3: lock
4: if current Time _ lastTime > THRESHOLD
or success ¼¼ false then
5: success false
6: lastTime current Time
7: for pro 2 providers do
8: updatepro:url; P
9: end for
10: flushP
11: success true
12: end if
13: unlock
14: data groups getDataGroupsRul:cfg
15: X getAllPlacementConfigproviders
16: for dg 2 data groups do
17: min MAXFLOAT
18: res NULL
19: csts getConstraintsRul:cfg; dg
20: opts getOptReqAndWeightRul:cfg; dg

21: stts getStatisticsSys:cfg; dg
22: for X s 2 X do
23: if subtoConstraintsðX s; csts; stts; Pthen
24: insertcands;X s
25: end if
26: end for
27: bp getBestPointcands; opts; stts; P
28: mp getMaximumPointcands; opts; stts; P
29: for X s 2 cands do
30: ED s calEDX s; opts; stts; bp; mp; P
31: if EDs < min then
32: min ED s
33: res X s
34: end if
35: end for
36: vals

VIII. ADVANTAGES

- No Vender Lock In
- Data is more secured
- No data lost while load

IX. CONCLUSION

This paper presents Triones, a systematic model to formulate and optimize data placement in multi-cloud storage by using erasure coding. As a systematic approach, Triones tries its best to avoid ad-hoc ways of randomly choosing data placement configurations. It uses non-linear programming to define the problem of data placement optimization. In this model, quantifiable factors under consideration can be expressed in the inequalities of constraints as well as being put in the objective function. We apply Euclidean distance measure through geometric space abstraction for the objective function to calculate the optimization results. In this way, complex requirements that are not considered in previous works can be easily included in Triones. Furthermore, new factors and requirements can be adopted in the model and optimized by the same means. Triones helps system or application developers to achieve the features of the multi-cloud storage in an optimized way with reasonable overhead.

X. FUTURE ENHANCEMENT

In our future work a model and a methodology that allow a tenant to estimate the costs of plain and encrypted cloud database services even in the case of workload and cloud price variations in a mid-term horizon. By instantiating the model with actual cloud provider prices, we can determine the encryption and adaptive encryption cost of data confidentiality. From the research point of view, it would be also interesting to evaluate the proposed or alternative architectures under different threat model hypotheses.

REFERENCE

[1] S. Mu, K. Chen, P. Gao, F. Ye, Y. Wu, and W. Zheng, “mlibcloud:Providing high available and uniform accessing to multiple cloudstorages,” in Proc.ACM/IEEE 13th Int. Conf. Grid Comput., 2012,pp. 201–208.
[2]Scalia: An Adaptive Scheme for EfficientMulti-Cloud StorageThanasis G. Papaioannou, Nicolas Bonvin and Karl AbererSchool of Computer and Communication SciencesEcole Polytechnique F´ed´erale de Lausanne (EPFL)1015 Lausanne, Switzerland
firstname.lastname@epfl.ch
[3] H. Abu-Libdeh, L. Prince house and H. Weatherspoon, ”RACS: A Casefor Cloud Storage Diversity”, in Proc. of SOCC, Indianapolis, USA,2010.

[4] H. Weatherspoon and J. Kubiatowicz, ”Erasure Coding Vs. Replication: A Quantitative Comparison”, in Revised Papers from IPTPS’01,Springer-Verlag, London, UK, 2002.
[5] K. D. Bowers and A. Juels and A. Opera, ”HAIL: a high-availabilityand integrity layer for cloud storage”, in Proceedings of the 16th ACMconference on Computer and communications security, Chicago, Illinois,USA, 2009
[6]High Availability in DHTs: Erasure Coding vs. Replication Author: Rodrigo Rodrigues and Barbara Liskov Year: 2011
[7] D. Abramson, J. Giddy, and L. Kotler, ”High Performance ParametricModeling with Nimrod/G: Killer Application for the Global Grid?”, inProc. of the IPDPS, 2000.
[8] J. Brunelle, P.Hurst, J.Huth, L.Kang, C.Ng, D.C.Parkes, M.Seltzer,J.Shank S.Youssef, ”Egg: an extensible and economics-inspired Open grid computing platform”, in Proc. of the Grid Economics Workshop(GECON), 2006.