Imperial Journal of Interdisciplinary Research (IJIR)
Vol-3, Issue-4, 2017
ISSN: 2454-1362, http://www.onlinejournal.in

Cost Optimization for Online Social Sites on Geo-Distributed Clouds

Prasad Bhamre[1]; Namrata Nikam[2]; Dattatray Badhe[3];
[1-2-3]Student, Department Of Computer Engineering ,
Yadavrao Tasgaokar Institute of Engineering & Technology, Karjat(India)

Abstract: *Modern cloud infra structures are geo-distributed. Geo distribution offers many advantages but can increase the total cloud capacity required. geo-distributed cloud services is a trend in cloud computing that, by spanning multiple data centers at different geographical locations, can provide a cloud platform with much larger capacities. Such a geo-distributed cloud is ideal for supporting large-scale social media and other applications with dynamic contents and demands. Here challenge for online services provider is optimizing the monetary cost spent in using cloud resources while considering other important requirements, including satisfactory quality of service (QoS) and data availability to online services users. This paper proposes efficient proactive algorithms for dynamic, optimal scaling of a social media application in a geo-distributed cloud*

Key Words**:** *cloud computing, online services, optimization model, performance, geo-distributed clouds*
.

1. Introduction

Most large online services are geo-distributed. Cloud infrastructures make it easy for any application to become Geo-distributed. Many techniques have been developed for geo-distributed application design such as distributed failure detection and request re-routing, geo-distributed storage consistency management, data placement. Most existing cloud systems—e.g., Amazon Elastic Compute Cloud (EC2)[1][2] and Simple Storage Service (S3), Microsoft Azure. A geo-distributed federated cloud is ideal for supporting large-scale social media streaming applications. Social network applications (e.g., Facebook, Twitter, Foursquare) are famous over the Internet today, and they are combined with conventional applications, such as multimedia streaming, to produce new social media applications, e.g., YouTube-like sites. Compared to traditional Internet video services, social media applications feature highly dynamic contents and demands since most of their videos are short, e.g., several minutes, a latency of more than a few tens of seconds would be intolerable to a viewer. It is therefore challenging to design and scale a social media application cost-effectively.

1.

Our system is to optimize the cost for the dynamic online services on multiple geo-distributed clouds over consecutive. time periods while meeting pre-defined QoS and data availability requirements. We first model the cost, the QoS, and the data availability of the online services which service upon clouds. Our cost model approximates the total cost of online services over consecutive time periods when the online services like social sites are large in user population but moderate in growth, enabling us to achieve the optimization of the total cost by independently optimizing the cost of each period. Our QoS model links the QoS with online services users' data locations among clouds. For every user, all clouds available are sorted in terms of a certain quality metric. Our data availability model relates with the minimum number of replicas maintained by each online services user. We then formulate the cost optimization problem that considers QoS and data availability requirements.

1.1 Existing System

Existing work on Online services providing either pursues least cost in a single site without the QoS concern as in the geo-distribution case or aims for least inter-data-center traffic in the case of multiple data centers without considering other dimensions of the service, e.g., data availability. More importantly, the models in all such work do not capture the monetary cost of resource usage and thus cannot fit the cloud scenario. Most optimization research on multi cloud and multi-data-center services is not for online services.

1.2 Literature Survey

A. **Multilevel algorithms for partitioning power-law graphs**

Graph partitioning is an enabling technology for parallel processing as it allows for the effective decomposition of unstructured computations whose data dependencies correspond to a large sparse and irregular graph. Even though the problem of computing high-quality partitioning of graphs arising in scientific computations is to a large extent wellunderstood, this is far from being true for emerging HPC applications whose underlying computation involves graphs whose degree distribution follows a power law curve. This paper presents new multilevel graph partitioning algorithms that are specifically designed for partitioning such graphs. It presents new clustering-based coarsening schemes that identify and collapse together groups of vertices that are highly connected. An experimental evaluation of these schemes on 10 different graphs shows that the proposed algorithms consistently and significantly outperform existing state-of-the-art approaches.

B. **Volley: Automated data placement for geo-distributed cloud services**

As cloud services grow to span more and more globally distributed datacenters, there is an increasingly urgent need for automated mechanisms to place application data across these datacenters. This placement must deal with business constraints such as WAN bandwidth costs and datacenter capacity limits, while also minimizing user-perceived latency. The task of placement is further complicated by the issues of shared data, data inter-dependencies, application changes and user mobility. We document these challenges by analyzing month-long traces from Microsoft's Live Messenger and Live Mesh, two large-scale commercial cloud services.

We present Volley[11], a system that addresses these challenges. Cloud services make use of Volley by submitting logs of datacenter requests. Volley analyzes the logs using an iterative optimization algorithm based on data access patterns and client locations, and outputs migration recommendations back to the cloud service.

To scale to the data volumes of cloud service logs, Volley is designed to work in SCOPE , a scalable MapReduce-style platform; this allows Volley to perform over 400 machine-hours' worth of computation in less than a day. We evaluate Volley on the month-long Live Mesh trace, and we find that, compared to a state-of-the-art heuristic that places data closest to the primary IP address that accesses it, Volley simultaneously reduces datacenter capacity skew by over 2×, reduces inter-datacenter traffic by over 1.8× and reduces 75th percentile user-latency by over 30%.

C. **Characterizing user behavior in online social networks**

Understanding how users behave when they connect to social networking sites creates opportunities for better interface design, richer studies of social interactions, and improved design of content distribution systems. In this paper, we present a first of a kind analysis of user workloads in online social networks. Our study is based on detailed clickstream data, collected over a 12-day period, summarizing HTTP sessions of 37,024 users who accessed four popular social networks: Orkut, MySpace, Hi5, and LinkedIn. The data were collected from a social network aggregator website in Brazil, which enables users to connect to multiple social networks with a single authentication. Our analysis of the clickstream data reveals key features of the social network workloads, such as how frequently people connect to social networks and for how long, as well as the types and sequences of activities that users conduct on these sites. Additionally, we crawled the social network topology of Orkut, so that we could analyze user interaction data in light of the social graph. Our data analysis suggests insights into how users interact with friends in Orkut, such as how frequently users visit their friends' or non-immediate friends' pages. In summary, our analysis demonstrates the power of using clickstream data in identifying patterns in social network workloads and social interactions. Our analysis shows that browsing, which cannot be inferred from crawling publicly available data, accounts for 92% of all user activities. Consequently, compared to using only crawled data, considering silent interactions like browsing friends' pages increases the measured level of interaction among users.

2. Proposed System

After going through the existing system and the literature survey we have figured out that there are a lot of problems in it. So, what we propose is a solution to develop system that can be useful in cost optimization for online services on geo distributed cloud. This system has four modules as follows

i. OSN System Construction Module

In the first module, we develop the Online Social Networking (OSN) system module. We build up the system with the feature of Online Social Networking. Where, this module is used for new user registrations and after registrations the users can login with their authentication. Where after the existing users can send messages to privately and publicly, options are built. Users can also share post with others. The user can able to search the other user profiles and public posts. In this module users can also accept and send friend requests.

With all the basic feature of Online Social Networking System modules is build up in the initial module, to prove and evaluate our system features. Clouds and OSN users are all geographically distributed. Without loss of generality, we consider the single-master–multi-slave paradigm.

ii. Modeling the Storage & Intercloud

Traffic Cost

In this module, we develop modeling the Storage and intercloud Traffic Cost of OSN, which is commonly abstracted as a social graph, where each vertex represents a user and each edge represents a social relation between two users. In this module we calculate the Storage Cost. A user has a storage cost, which is the monetary cost for storing one replica of her data (e.g., profile, statuses) in the cloud for one billing period. Similarly, a user has a traffic cost, which is the monetary cost during a billing period because of the intercloud traffic. As mentioned earlier, due to social locality, in our settings the intercloud traffic only involves writes (e.g., posting tweets, leaving comments). We do not consider intracloud traffic, no matter read or write, as it is free of charge. A user has a sorted list of clouds for the purpose of QoS

iii) **Modeling the Redistribution Cost**

An important part of our cost model is the cost incurred by the optimization mechanism itself, which we call the redistribution cost. We generally envisage that an optimization mechanism is devised to optimize the cost by moving data across clouds to optimum locations, thus incurring such cost. The redistribution cost is essentially the intercloud traffic cost, but in this paper we use the term intercloud traffic to specifically refer to the intercloud write traffic for maintaining replica consistency, and treat the redistribution cost separately

iv) **Approximating the Total Cost**

Consider the social graph in a billing period. As it may vary within the period, we denote the final steady snapshot of the social graph in this period, and the initial snapshot of the social graph at the beginning of this period. The storage cost in is for storing users' data replicas, including the data replicas of existing users and of those who just join the service in this period. The intercloud traffic cost in is for propagating all users' writes to maintain replica consistency. The redistribution cost is the cost of moving data across clouds for optimization; it is only incurred at the beginning of a period. There is also some underlying cost for maintenance.

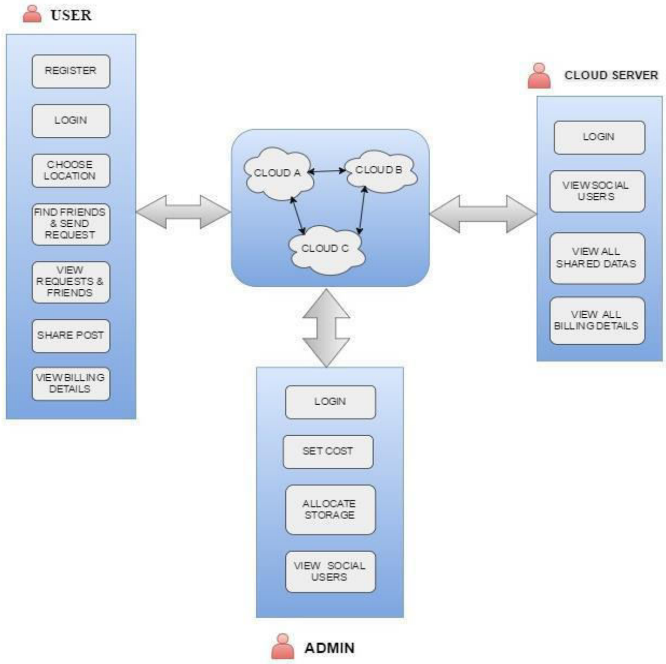

Fig -1: System Architecture

3. Conclusions

We propose to study the problem of optimizing the monetary cost spent on cloud resources when deploying online services over multiple geo-distributed clouds. On the basis of cost of online services data placement, quality of service (QoS) and data availability models, we present the optimization problem of minimizing the total cost while ensuring the QoS and the data availability.

4. Acknowledgement

This work is a part of graduation project done by students of Computer Engineering. We thank everyone who supported and motivated us. And special thanks to our guide Mrs. Nilima Nikam.

5. References

[1] Amazon EC2 Pricing https://aws.amazon.com/ec2/pricing/

[2] Case Studies. http://aws.amazon.com/solutions/case-studies/

[3] Data Center Map http://www.datacentermap.com/cloud.html/

[4] Pricing Overview: Microsoft Azure. http://www.Windowsazure.com/en-us/pricing/details/.

[5] K. Schloegel, G. Karypis, and V. Kumar. Wavefront

diffusion and lmsr: Algorithms for dynamic repartitioning of adaptive meshes. IEEE Transactions on Parallel and Distributed Systems, 12(5):451– 466, 2001

[6] A. Abou-Rjeili and G. Karypis. Multilevel algorithms for partitioning power-law graphs. In IPDPS, 2006.

[7] L. Jiao, J. Li, T. Xu, and X. Fu, “Cost optimization for online social networks on geo-distributed clouds,” in Proc. IEEE ICNP,
2012, pp. 1–10.

[8] Y. Yang, Q. Chen, and W. Liu, “The structural evolution of an online discussion network,” Physica A, Statist. Mech. Appl., vol.389, no. 24, pp. 5871–5877, 2010.

[9] Y. Wu, C. Wu, B. Li, L. Zhang, Z. Li, and F. C. M. Lau, “Scaling social media applications into geo-distributed clouds,” in Proc. IEEE INFOCOM, 2012,pp.684–692.

[10] A. E. Mislove, “Online social networks: Measurement, analysis, and applications to distributed information systems,” Ph.D. dissertation, Rice University, Houston, TX, USA, 2009

[11]S. Agarwal, J. Dunagan, N. Jain, S. Saroiu, A. Wolman, and H. Bhogan. Volley: Automated data placement for geo-distributed cloud services. In NSDI, 2010.