Minimizing the memory Cost in Clouds using Data Staging Algorithm

Prof.Sanchika Bajpai[1], Sujata Yadav[2],Reshma Ruke[3], Trupti Mahadik[4]

Department of Computer Engineering,BSIOTR,Pune.India.

Abstract— **In this paper, we study about efficiently achieving data staging and caching on a set of advantage sites with a less cost in a cloud system. In this we focus on the problem of prices of high bandwidth which is charged by cloud providers for uploads and downloads of customer data. Unlike the traditional research, we do not design to identify the access patterns to facilitate the future requests. Instead, with information probably known in advance, while minimizing the monetary costs for caching and transmitting the data items requested by user. our goal for designing this paper is to efficiently stage the shared data items to predetermined sites at promoter time instants to align with the patterns. In this we create the single or multiple copies of data items and store on the some vantage site. We present a bulk transfer system that opportunistically exploits the excess capacities of network links to deliver bulk content cheaply and efficiently. When the ratio of transmission cost and caching cost is low, a single copy of each data item can efficiently serve to all the requested user. In multi copy situation, our main focus on the tradeoff between the transmission cost and storing cost by controlling the upper bounds of transmissions and copies .The upper bound can be on per-item basis or on all-item basis. Based on dynamic programming techniques , we present efficient optimal solutions to all these cases provided that the upper bound is poly nomially surronded by the number of service requests and the number of distinct data items.**

Index Terms— **Cloud computing, data placement and migration, data staging and caching, resource constraints.**

INTRODUCTION

Now a days the data accessibility demand on clouds is increases ,data availability maximization have some important problem to maintain high-reliability and time-bounded service expectations in clouds. For example, In the shortest possible time, one of the pressing needs by the cloud service providers (CSPs) is to efficiently serve the needs of the user requests that demand single or multiple data items. Thus, the problem of making the requested data making available to the users becomes an important issue for CSPs to guarantee high-quality services because of increasing the population of users on cloud. An approach to increasing data availability is to stage the requested data to some vantage sites and cache the data for a period of time. so that the quality of service(QOS)like latency minimization, network traffic reduction etc. for user's future accesses can be greatly improved. We can called to this combined functionality as data staging. An worthy of imitation scenario is a mobile user who may want a service to suggest probable alternative routes to her destination depending on current traffic patterns. This service requires not only making the queried data highly available at vantage nodes along her current path but also staging the data at key sites on probable paths based on her navigation history. From this example, it is assume that ,data staging could effectively facilitate the service accesses. However ,Due to the maximum cost we can not take the advantage of shared data accesses on cloud.

In this paper, we study data staging problem by holding the dynamic programming (DP) techniques to replicate, migrate and cache the shared data items in cloud systems with or without some practical resource constraints in an efficient way while minimizing the monetary cost for transmitting and caching the data items .Monetary cost is our first interest which is a very flexible concept to reflect the qualities of various network features such as network bandwidth, link latency, and storage utilization. on the other hand, the provision of the resources in cloud systems are usually based on pay-as-you-go fashion. Thus effective use of the platforms within budget constraint is always the user's concern. Due to the optimality, our solutions are unique over other methods to provide the cloud-based services with the flexibility that they cannot decide not only the time of each data item to be cache data from some vantage sites but also make a tradeoff between transmission cost and storing cost to meet the constraints imposed by the underlying Infrastructure as a Service Providers (IaaSPs), information item owners or Cloud Service Provider's(CSP)'budget.

I. PROPOSED METHOD

Data staging algorithm is not only used to figure out the user's access patterns i.e., a sequence of requests, each being made by certain node at a predicted time instant but also required to replicate, migrate or store the included data items across the network to optimize some performance metric for future accesses at a particular time instants..

First we propose the service model , to get better the network capacity. Our model contain two classes: the standard best-effort service used for latency-sensitive traffic, and a bulk service that opportunistically uses the left over capacity for latency-tolerant bulk content. Second, we present a networked system called NetEx that enables an ISP to provide a bulk transfer service [1]. Instead ,with a such kind of information known in progress, our aim is to efficiently migrate, replicate, or cache the several requested data items

defined by the access pattern in a fully connected network with or without resource constraints .Tthe cost in the data staging and caching accounts for a part of the total cost, minimizing it can accordingly minimize the total cost. With the cost of the second phase reducing, the first phase of the algorithm could have more chances to adjust the frequency of running the ledge algorithms to improve the prediction accuracy so that the overall cost could be furthermore reduced.

II. COST MODELS

The cost model adopted in our research could be heterogeneous or homogeneous in the senses that whether or not the transmission costs are identical and caching costs at all sites are also identical. As our algorithms are mainly designed for CSPs who usually demand the infrastructure services from an IaaSPs (e.g., Amazon AWS, GoGrid,Flexiscale and Mosso), in this paper we are particularly interested in the situation when the homogeneous cost model is employed due to two reasons: First, the borrowed infrastructure for a exacting service is always organized as a subset of homogeneous resources to entail the hosted applications to meet its Service Level Agreement (SLA)targets. This results is common in *homogeneous* computation and communication in the clouds.

(1).Homogeneous computation: IaaSPs commonly provide users with a set of different virtual machine types, each of which has different resource capacities in terms of CPU ability, RAM range, disk I/O bandwidth etc. The performance of different types of virtual machines are obviously heterogeneous. However, the performance of multiple virtual machines of the same type which usually host a particular service is practically similar.

(2).Homogeneous communication: The current topologies of data center networks are in general structured as either two- or three-level trees of switches or routers with an low-bandwidth *edge* tier at the leaves and high-bandwidth fat-tree at the *top* of the tree, approximately important to homogeneous communication in nature

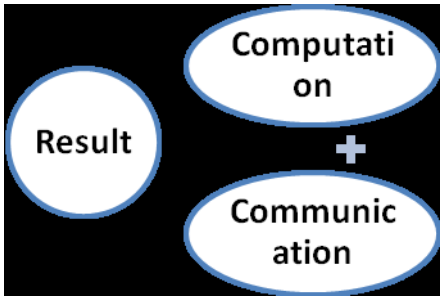

Figure1. Homogeneous communication and computation

Second, as both the transmission cost rate and caching cost rate are resolute by the IaaSP it is not likely for them to offer a heterogeneous cost model as it could create difficulties in public clouds since at present only a small number of IaaSPs are disposed to discovery some low-level information about the containers and sub networks to their users.

III. PROBLEM FORMULATION

The algorithm design, it is not always sufficient to reflect the reality where some association rules may exist among the user accesses and caching or transmitting the data items in bulk could be cheaper than in each item individually .Suppose there are k distinct shared data item initially stored at one node, say p_1 and later migrated ,replicated ,and cached in a fully connected network with m nodes (p_1, $p_{2, ...,}$ p_m) to serve a sequence of requests$\sigma \triangleq \sigma_1\sigma_2...\sigma_n$ in which each $\sigma_i \triangleq (t_i, p_i, R_i)$ $1 \leq i \leq n$is specified by the predicted access pattern and represents a request made for a data subset R_i by node p_i at time t_i. We therefore have the complete knowledge of all such information as an input to our algorithms[2]. Further, for simplicity we also suppose that there exists only one request per phase. To assure a request for a particular data item, we describe the following primitive operations to perform on the cached information items, which may occupy caching and transmission costs:

1. *Retainment* : At a node p_u cache the data item from time t_u to t_v by paying $(t_v\text{-}t_u)S_u$, S_u is rate cost of caching at node p_u,$1 \leq u \leq m$.
2. *Migration*: the data item are move from a node p_u to anode p_v at a cost equal to the distance C_{uv}.
3. *Replication:* at a cost of $C_{uv,}$ the items are copy to the request node p_v from a node p_u.
4. *Excursion:* the request is satisfy at a node p_v by using the copy at a node p_u without migration at a cost of E_{uv}.
5. *Creation/deletion*: without incurring any cost, create/delete the selected copies at some nodes.

Cost of replication and excursion are not identical. especially based on the amount and duration that clients use, the storage, are always charged not including their initialization and finalization. However, in cloud systems such as GAE cloud computing, Amazon web services, and Windows navy Platform, there sources. Thus, we follow this principle in our model to allow $E_{uv} = C_{uv}$ for making any pair of p_u and p_v in the network.

IV. CONCLUSION

In this paper, we studied the problem of staging a set of distinct data items in a fully connected network to facilitate cloud-based services with minimum cost. we search into the optimal staging strategies based on the cost models in the paper [3] to minimize the total staging cost. We also considered some practical constraints on this problem in terms of the maximum number of transmissions and copies.

We achieved an well-organized optimal solution via dynamic programming to the position when the numbers of transmissions and copies are limitless within the time complexity of O (kmn^2). When S > C, this algorithm as well ensure an best single-copy algorithm within the complexity of O (kn^2). For random S and C, our results show that when C/S is low, a single copy of each data item can efficiently provide all the user request sequence.

In this paper, we assumed that the accesses to the k distinct items are independent with each other and the cost model is defined on per-item basis. Although the assumptions can simplify the algorithm design, it is not always sufficient to reflect the reality where some associate on rules may exist among the user accesses and caching or transmitting the data

items in bulk could be cheaper than In each item individually. We will study these issues in the future work. Considering this problem under the storage constraints is another interesting problem. As each node is equipped with fixed storage size, the caching cost would be time dependent and not increase linearly with caching time .Therefore, solutions to the problem with this constraint is more practical in reality, and thus worthwhile to study in the future.

REFERENCES

[1] Kalaiselvi.S1; Newlin Rajkumar.M2; " Achieving Minimum Transferring And Storage Cost On Data Staging Algorithms For Shared Data Accesses In Clouds", PG Scholar, Regional Centre of Anna University, Coimbatore, India1

[2] Yang Wang,"On Data Staging Algorithms for SharedData Accesses in Clouds",Bharadwaj Veeravalli, Senior Member, IEEE, and Chen-Khong Tham

[3] Bolan B. Veeravalli, "Network Caching Strategies for a Shared Data Distribution for a Predefined Service Demand Sequence," IEEETrans. Knowledge and Data Eng., vol. 15, no. 6, pp. 1487-1497, Nov.2003.

BIOGRAPHIES

Prof. Sanchika Bajpai
Department of computer,
BSIOTR,
Pune, Maharashtra.
India.

Sujata Yadav
Department of computer,
BSIOTR,
Pune, Maharashtra.
India.

Reshma Ruke
Department of computer,
BSIOTR,
Pune, Maharashtra.
India.

Trupti Mahadik
Department of computer,
BSIOTR,
Pune, Maharashtra.
India.

