International Journal for Modern Trends in Science and Technology
Volume: 03, Issue No: 09, September 2017
ISSN: 2455-3778
http://www.ijmtst.com

A Profit Enhancement Scheme with assured Quality of Service in Cloud Computing

Sunkara Tejaswini[1] | CH Srinivas Rao[2]

[1]Department of CSE, Amrita Institute of Science And Technology, Paritala, Kanchikacherla, Krishna district, AP, India.
[2]Professor, Department of CSE, Amrita Institute of Science And Technology, Paritala, Kanchikacherla ,Krishna district, AP, India.

To Cite this Article
Sunkara Tejaswini and Ch Srinivas Rao, "A Profit Enhancement Scheme with assured Quality of Service in Cloud Computing", *International Journal for Modern Trends in Science and Technology*, Vol. 03, Issue 09, September 2017, pp.-70-75.

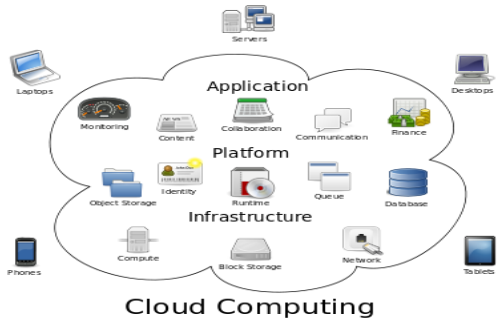
ABSTRACT

In today life quality is the most challenging issue. As an effective and efficient way to provide computing resources and services to customers on demand, cloud computing has become more and more popular. From cloud service provider's perspective, profit is one of the most important considerations, and it is mainly determined by the configuration of a cloud service platform under given market demand. However, a single long-term renting scheme is usually adapted to configure a cloud platform, which cannot guarantee the service quality but leads to serious resource waste. In this paper, a double resource renting scheme is designed at first in which short-term renting and long-term renting are combined aiming at the existing issues. This double renting scheme can effectively guarantee the quality of service of all requests and reduce the resource waste greatly. Secondly, a service system is considered as an M/M/m+D queuing model and the performance indicators that affect the profit of our double renting scheme are analyzed, e.g., the average charge, the ratio of requests that need temporary servers, and so forth. Thirdly, a profit maximization problem is formulated for the double renting scheme and the optimized configuration of a cloud platform is obtained by solving the profit maximization problem. Finally, a series of calculations are conducted to compare the profit of our proposed scheme with that of the single renting scheme. The results show that our scheme can not only guarantee the service quality of all requests, but also obtain more profit than the latter.

Keywords: *Cloud computing, guaranteed service quality, multiserver system, profit maximization, queuing model, service-level agreement, waiting time..*

I. INTRODUCTION

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers.



Structure of Cloud Computing

The cloud computing uses networks of large groups of servers typically running low-cost consumer PC technology with specialized connections to spread data-processing chores across them. This shared IT infrastructure contains large pools of systems that are linked together. Often, virtualization techniques are used to maximize the power of cloud computing.

II. THE MODELS

In this section, we first describe the three-tier cloud computing structure. Then, we introduce the related models used in this paper, including a multiserver system model, a revenue model, and a cost model.

A. A Cloud System Model

The cloud structure (see Fig. 1) consists of three typical parties, i.e., infrastructure providers, service providers and customers. This three-tier structure is used commonly in existing literatures [2, 6, 10].

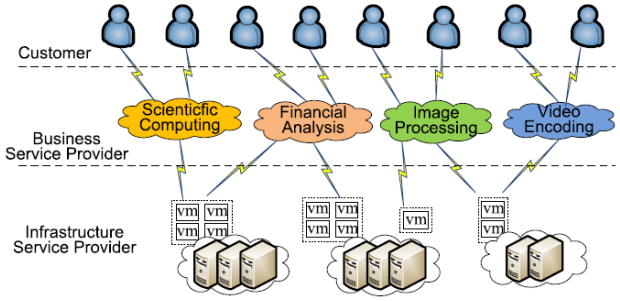


Fig. 1: The three-tier cloud structure.

In the three-tier structure, an infrastructure provider the basic hardware and software facilities. A service provider rents resources from infrastructure providers and prepares a set of services in the form of virtual machine (VM). Infrastructure providers provide two kinds of resource renting schemes, e.g., long-term renting and short-term renting. In general, the rental price of long-term renting is much cheaper than that of short-term renting. A customer submits a service request to a service provider which delivers services on demand. The customer receives the desired result from the service provider with certain service-level agreement, and pays for the service based on the amount of the service and the service quality. Service providers pay infrastructure providers for renting their physical resources, and charge customers for processing their service requests, which generates cost and revenue, respectively. The profit is generated from the gap between the revenue and the cost.

B. A Multiserver Model

In this paper, we consider the cloud service platform as a multiserver system with a service request queue. Fig. 2 gives the schematic diagram of cloud computing .

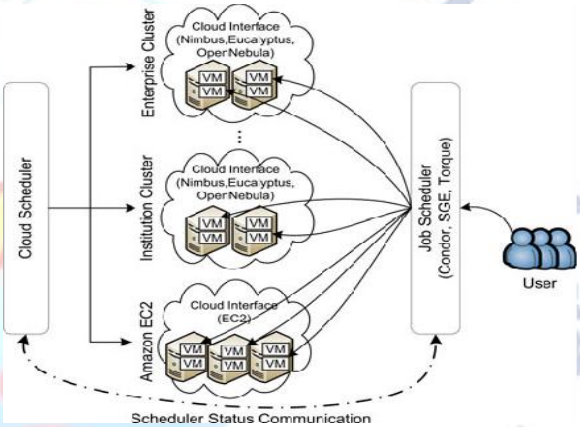


Fig. 2: The schematic diagram of cloud computing

All jobs are scheduled by the job scheduler and assigned to different VMs in a centralized way. Hence, we can consider it as a service request queue. For example, Condor is a specialized workload management system for compute-intensive jobs and it provides a job queuing mechanism, scheduling policy, priority scheme, resource monitoring, and resource management. Users submit their jobs to Condor, and Condor places them into a queue, chooses when and where to run them based upon a policy

Hence, it is reasonable to abstract a cloud service platform as a multiserver model with a service request queue, and the model is widely adopted in existing literature [2, 11].

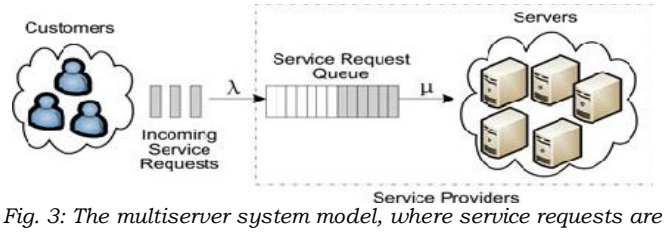


Fig. 3: The multiserver system model, where service requests are first placed in a queue before they are processed by any servers.

Assume that the multiserver system consists of m long-term rented identical servers, and it can be scaled up by temporarily renting short-term servers from infrastructure providers. The servers in the system have identical execution speed s (Unit: billion instructions per second). In this paper, a multiserver system excluding the short-term servers is modeled as an M/M/m queuing system as follows (see Fig. 3).

C. Revenue Modeling

The revenue model is determined by the pricing strategy and the server-level agreement (SLA). In this paper, the usage-based pricing strategy is adopted, since cloud computing provides services to customers and charges them on demand. The SLA is a negotiation between service providers and customers on the service quality and the price. Because of the limited servers, the service requests that cannot be handled immediately after entering the system must wait in the queue until any server is available. However, to satisfy the quality-of-service requirements, the waiting time of each service request should be limited within a certain range which is determined by the SLA. The SLA is widely used by many types of businesses, and it adopts a price compensation mechanism to guarantee service quality and customer satisfaction. For example, China Post gives a service time commitment for domestic express mails. It promises that if a domestic express mail does not arrive within a dead-line, the mailing charge will be refunded. The SLA is also adopted by many real world cloud service providers such as Rackspace, Joyent, Microsoft Azure, and so on. Taking Joyent as an example, the customers order Smart Machines, Smart Appliances, and/or Virtual Machines from Joyent, and if the availability of a customer's services is less than 100%, Joyent will credit the customer 5% of the monthly fee for each 30 minutes of downtime up to 100% of the customer's monthly fee for the affected server. The only difference is that its performance metric is availability and ours is waiting time.

D. Cost Modeling

The cost of a service provider consists of two major parts, i.e., the rental cost of physical resources and the utility cost of energy consumption. Many existing research such as [11] only consider the power consumption cost. As a major difference between their models and ours, the resource rental cost is considered in this paper as well, since it is a major part which affects the profit of service providers. A similar cost model is adopted in [2]. The resources can be rented in two ways, long-term renting and short-term renting, and the rental price of long-term renting is much cheaper than that of short-term renting. This is reasonable and common in the real life. In this paper, we assume that the long-term rental price of one server for unit of time is β (Unit: cents per second) and the short-term rental price of one server for unit of time is γ (Unit: cents per second), where $\beta < \gamma$.

III. A QUALITY-GUARANTEED SCHEME

The traditional single resource renting scheme cannot guarantee the quality of all requests but wastes a great amount of resources due to the uncertainty of system workload. To overcome the weakness, we propose a double renting scheme as follows, which not only can guarantee the quality of service completely but also can reduce the resource waste greatly.

A. The Proposed Scheme

In this section, we first propose the Double-Quality-Guaranteed (DQG) resource renting scheme which com-bines long-term renting with short-term renting. The main computing capacity is provided by the long-term rented servers due to their low price. The short-term rented servers provide the extra capacity in peak period. The detail of the scheme is shown in Algorithm 1.

The proposed DQG scheme adopts the traditional FCFS queueing discipline. For each service request entering the system, the system records its waiting time. The requests are assigned and executed on the long-term rented servers in the order of arrival times. Once the waiting time of a request reaches D, a temporary server is rented from infrastructure.

Algorithm 1 Double-Quality-Guaranteed (DQG) Scheme

1: A multiserver system with m servers is running and waiting for the events as follows.
2: A queue Q is initialized as empty
3: Event – A service request arrives
4: Search if any server is available
5: if true then
6: Assign the service request to one available server
7: else

8: Put it at the end of queue Q and record its waiting time
9: end if
10: End Event
11: Event – A server becomes idle
12: Search if the queue Q is empty
13: if true then
14: Wait for a new service request
15: else
16: Take the first service request from queue Q and assign`n it to the idle server
17: end if
18: End Event
19: Event – The deadline of a request is achieved
20: Rent a temporary server to execute the request and release the temporary server when the request is completed
21: End Event

Hence, the revenue of the service provider increases. However, the cost increases as well due to the temporarily rented servers. Moreover, the amount of cost spent in renting temporary

In the three-tier structure, a cloud service provider serves customers' service requests by using a multiserver system which is rented from an infrastructure provider. servers is deter-mined by the computing capacity of the long-term rented multiserver system. Since the revenue has been maximized using our scheme, minimizing the cost is the key issue for profit maximization. Next, the tradeoff between the long-term rental cost and the short-term rental cost is considered, and an optimal problem is formulated in the following to get the optimal long-term configuration such that the profit is maximized.

B. The Profit Optimization Problem

It is known that part of requests need temporary servers to serve, so that their quality can be guaranteed. Denoted by pext(D) the steady-state probability that a request is assigned to a temporary server, or put differently, pext(D) is the long-run fraction of requests whose waiting times exceed the deadline D. pext(D) is different from FW (D). In calculating FW (D), all service requests, whether exceed the deadline, will be waiting in the queue. However, in calculating pext(D), the requests whose waiting times are equal to the deadline will be assigned to the temporary servers, which will reduce the waiting time of the following requests. In general, pext(D) is much less than FW (D). We can known that pext(D) is:

$pext(D) = (1 - \rho)(1 - FW(D))$

$1 - \rho(1 - FW(D))$

The profit of a service provider in one unit of time is obtained as

Profit = Revenue − Clong − Cshort, where Revenue = λar,

IV. INPUT AND OUTPUT REPRESENTATION

A. Input Design

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things: What data should be given as input? How the data should be arranged or coded? The dialog to guide the operating personnel in providing input methods for preparing input validations and steps to follow when error occur.

B. Output Design

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.
2. Select methods for presenting information.
3.Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives. Convey information about past activities, current status or projections of the Future. Signal important events, opportunities, problems, or warnings. Trigger an action. Confirm an action.

V. IMPLEMENTATION OF MODULES

A. Cloud Computing

Cloud computing describes a type of outsourcing of computer services, similar to the way in which the supply of electricity is outsourced. Users can simply use it. They do not need to worry where the electricity is from, how it is made, or transported. Every month, they pay for what they consumed. The idea behind cloud computing is similar: The user can simply use storage, computing power, or specially crafted development environments, without having to worry how these work internally. Cloud computing is usually Internet-based computing. The cloud is a metaphor for the Internet based on how the internet is described in computer network diagrams; which means it is an abstraction hiding the complex infrastructure of the internet. It is a style of computing in which IT-related capabilities are provided “as a service”, allowing users to access technology-enabled services from the Internet ("in the cloud") without knowledge of, or control over the technologies behind these servers.

B. Queuing Model

We consider the cloud service platform as a multiserver system with a service request queue. The clouds provide resources for jobs in the form of virtual machine (VM). In addition, the users submit their jobs to the cloud in which a job queuing system such as SGE, PBS, or Condor is used. All jobs are scheduled by the job scheduler and assigned to different VMs in a centralized way. Hence, we can consider it as a service request queue. For example, Condor is a specialized workload management system for compute intensive jobs and it provides a job queueing mechanism, scheduling policy, priority scheme, resource monitoring, and resource management. Users submit their jobs to Condor, and Condor places them into a queue, chooses when and where to run them based upon a policy. An M/M/m+D queueing model is build for our multiserver system with varying system size. And then, an optimal configuration problem of profit maximization is formulated in which many factors are taken into considerations, such as the market demand, the workload of requests, the server-level agreement, the rental cost of servers, the cost of energy consumption, and so forth. The optimal solutions are solved for two different situations, which are the ideal optimal solutions and the actual optimal solutions.

C. Business Service Providers Module

Service providers pay infrastructure providers for renting their physical resources, and charge customers for processing their service requests, which generates cost and revenue, respectively. The profit is generated from the gap between the revenue and the cost. In this module the service providers considered as cloud brokers because they can play an important role in between cloud customers and infrastructure providers, and he can establish an indirect connection between cloud customer and infrastructure providers.

D. Infrastructure Service Provider Module

In the three-tier structure, an infrastructure provider the basic hardware and software facilities. A service provider rents resources from infrastructure providers and prepares a set of services in the form of virtual machine (VM). Infrastructure providers provide two kinds of resource renting schemes, e.g., long-term renting and short-term renting. In general, the rental price of long-term renting is much cheaper than that of short-term renting.

E. Cloud Customers

A customer submits a service request to a service provider which delivers services on demand. The customer receives the desired result from the service provider with certain service-level agreement, and pays for the service based on the amount of the service and the service quality.

VI. CONCLUSION

Maximize the profit of service providers, this paper has proposed a novel Double-Quality-Guaranteed (DQG) renting scheme for service providers. This scheme combines short-term renting with long-term renting, which can reduce the resource waste greatly and adapt to the dynamical demand of computing capacity. An M/M/m+D queueing model is build for our multiserver system with varying system size. And then, an optimal configuration problem of profit maximization is formulated in which many factors are taken into considerations, such as the market

demand, the workload of requests, the server-level agreement, the rental cost of servers, the cost of energy consumption, and so forth. The optimal solutions are solved for two different situations, which are the ideal optimal solutions and the actual optimal solutions. In addition, a series of calculations are conducted to compare the profit obtained by the DQG renting scheme with the Single-Quality-Unguaranteed (SQU) renting scheme. The results show that our scheme outperforms the SQU scheme in terms of b.

REFERENCES

[1] A. Fiat and M. Naor, "Broadcast Encryption," in Proc. Crypto 1993, 1993, vol. LNCS 773, Lecture Notes in Computer Science, pp. 480- 491.

[2] I. Ingemarsson, D.T. Tang and C.K. Wong, "A Conference Key Distribution System," IEEE Transactions on Information Theory, vol. 28, no. 5, pp. 714-720, 1982.

[3] Q. Wu, Y. Mu, W. Susilo, B. Qin and J. Domingo-Ferrer, "Asymmetric Group Key Agreement," in Proc. Eurocrypt 2009, 2009, vol. LNCS 5479, Lecture Notes in Computer Science, pp. 153-170.

[4] http://en.wikipedia.org/wiki/PRISM %28surveillance program%29, 2014.

[5] Q. Wu, B. Qin, L. Zhang, J. Domingo-Ferrer and O. Farr`as, "Bridging Broadcast Encryption and Group Key Agreement," in Proc. Asiacrypt 2011, 2011, vol. LNCS 7073, Lecture Notes in Computer Science, pp. 143-160.

[6] D. H. Phan, D. Pointcheval and M. Strefler, "Decentralized Dynamic Broadcast Encryption," in Proc. SCN 2012, 2011, vol. LNCS 7485, Lecture Notes in Computer Science, pp. 166-183

[7] M. Steiner, G. Tsudik and M. Waidner, "Key Agreement in Dynamic Peer Groups," IEEE Transactions on Parallel and Distributed Systems, vol. 11, no. 8, pp. 769-780, 2000.

[8] A. Sherman and D. McGrew, "Key Establishment in Large Dynamic Groups Using One-way Function Trees," IEEE Transactions on Software Engineering, vol. 29, no. 5, pp. 444-458, 2003.

[9] Y. Kim, A. Perrig and G. Tsudik, "Tree-Based Group Key Agreement," ACM Transactions on Information System Security, vol. 7, no. 1, pp. 60-96, 2004.

[10] Y. Mao, Y. Sun, M. Wu and K.J.R. Liu, "JET: Dynamic Join-Exit- Tree Amortization and Scheduling for Contributory Key Management," IEEE/ACM Transactions on Networking, vol. 14, no. 5, pp. 1128-1140, 2006.

[11] C. Boyd and J.M. Gonz´alez-Nieto, "Round-Optimal Contributory Conference Key Agreement," in Proc. PKC 2003, 2003, vol. LNCS 2567, Lecture Notes in Computer Science, pp. 161-174.

[12] W.-G. Tzeng and Z.-J. Tzeng, "Round Efficient Conference Key Agreement Protocols with Provable Security," in Proc. Asiacrypt 2000, 2000, vol. LNCS 1976, Lecture Notes in Computer Science, pp. 614-627.