Computers and Electrical Engineering 58 (2017) 154–160

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

CLB: A novel load balancing architecture and algorithm for cloud services☆

Shang-Liang Chen, Yun-Yao Chen*, Suang-Hong Kuo

Institute of Manufacturing Information and Systems, National Cheng Kung University, Tainan, Taiwan, ROC

ARTICLE INFO

Article history:
Received 8 September 2014
Revised 10 January 2016
Accepted 12 January 2016
Available online 28 February 2016

Keywords:
CLB
Load balance
Cloud computing

ABSTRACT

Cloud services are widely used in manufacturing, logistics, digital applications, and document processing. Cloud services must be able to handle tens of thousands of concurrent requests and to enable servers to seamlessly provide the amount of load balancing capacity required to respond to incoming application traffic in addition to allowing users to obtain information quickly and accurately. In the past, researchers have proposed the use of static load balancing or server response times to evaluate load balancing capacity, a lack of which may cause a server to load unevenly. In this study, a dynamic annexed balance method is used to solve this problem. Cloud load balancing (CLB) takes into consideration both server processing power and computer loading, thus making it less likely that a server will be unable to handle excessive computational requirements. Finally, two algorithms in CLB are also addressed with experiments to prove the proposed approach is innovative.

1. Introduction

With the rapid development of the Internet, many vendors have started to provide cloud services. More and more services can be obtained in the cloud, and users do not have to do operations on a local computer. All operations are computed on the cloud. When a large number of users attempt to access cloud services, this often causes the server to fail to respond. Determining a method by which to provide users with timely and accurate responses is a subject worthy of advanced study. Several studies have been proposed to evaluate and to develop algorithms and load balancing methodologies for cloud-based applications. It is difficult for a server to deal effectively with the flow of information generated by all of the various enterprises attempting to access it. Excessive flow causes server overload with a subsequent loss of information. A server load balancing mechanism can disperse the transmission of information flow and data operations and can also reduce the probability of increased computational time and loss of information. When one server fails in the cloud, the cloud services can be transferred to another server. Services are therefore non-stop. The advantages of server load balance include:

1. Efficiency: improve the efficient use of the server and network bandwidth.
2. Reliability and Safety: improve the reliability and security of the server.
3. Scalability: improve the scalability of services.

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. W-H Hsieh.

* Corresponding author. Tel.: +886 62757575x61001
E-mail address: yewyewchen@gmail.com (Y.-Y. Chen).

http://dx.doi.org/10.1016/j.compeleceng.2016.01.029
0045-7906/

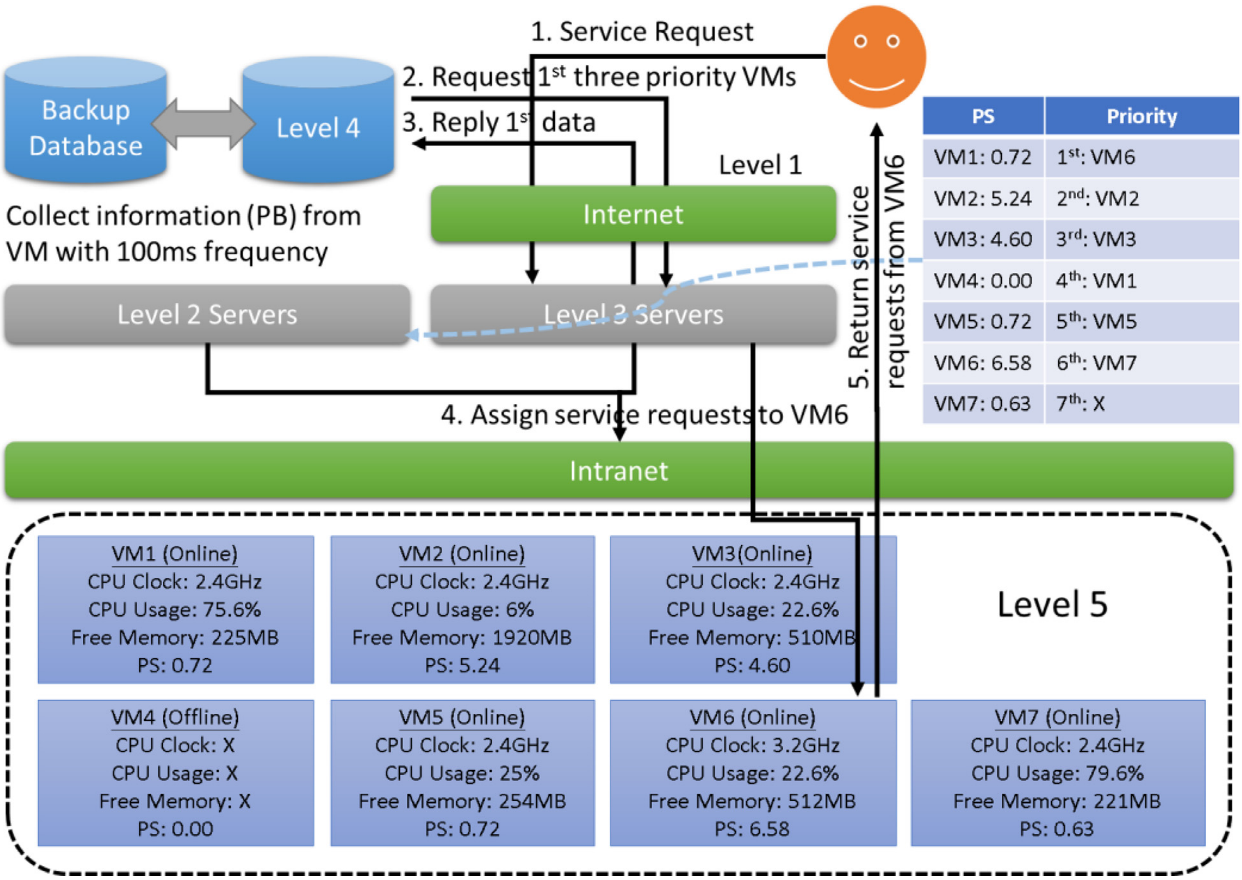

Fig. 1. Cloud load balancing architecture.

Currently, there are several types of load balancing techniques for cloud-based applications. A server load balancing method is divided into two types of hardware and software:

1. Hardware: Fourth layer network switches provide server redundancy and load balancing.
2. Software: Used to calculate server usage and to allocate resources.

Regardless of whether software or hardware is used, the load balancing method is divided into two types: dynamic and static, which are explained as follows:

1. Static: Used to deal with predictable processing loads in advance.
2. Dynamic: Used to deal with unpredictable processing loads. Based on network storage virtualization, the host and storage devices use fiber channel switches that link together, and all virtualization requests return to the network storage device. This method does not depend on the operating system.

Static load balancing uses the polling approach (also called the Round Robin approach). This approach is sequentially assigned to each host, as shown in Fig. 1. This method is simple and uses fewer resources, but is usually unable to detect the attached server, resulting in annexation or uneven distribution. Shadrach proposed an RTSLB algorithm based on a weighted metric and compared the results to three algorithms, including a random load distribution algorithm, a Round Robin load distribution algorithm and a competitive learning algorithm [1]. A comparison of existing load balancing techniques for current research hotspots was proposed by Raghava [2] and is listed as Table 1.

Based on previous studies on this topic [13], we propose a new paradigm for load balancing architecture and a new algorithm, which can be applied to both virtual web servers and physical servers. There was several discussions of load balancing techniques for hardware and software, including grid-based, microprocessor-based, wireless sensor network-based approaches [14–18]. In addition, we discuss more current approaches to load balancing techniques for cloud-based applications, which are listed in Table 2.

The rest of the paper is organized as follows: The introduction and earlier studies on cloud load balance are briefly summarized in Section 1. The methodology of the proposed segmentation is detailed in Section 2 to Section 4. Experimental results and discussions are shown in Section 5. Finally, conclusions are drawn in Section 6.