An Improved Dynamic Load-Balancing Model

Wenqian Shang, School of Computer Science, Communication University of China, Beijing, China & Beijing Key Laboratory of Big Data in Security & Protection Industry, Beijing, China
Di Liu, Communication University of China, Beijing, China & Beijing Key Laboratory of Big Data in Security & Protection Industry, Beijing, China
Ligu Zhu, Communication University of China, Beijing, China & Beijing Key Laboratory of Big Data in Security & Protection Industry, Beijing, China
Dongyu Feng, Communication University of China, Beijing, China & Beijing Key Laboratory of Big Data in Security & Protection Industry, Beijing, China

ABSTRACT

With the rapid development of big data and the national big data strategy is put forward, the Web server cluster facing more complex and severe challenges. The traditional load balancing algorithm has obvious limitations. This paper proposes a dynamic load-balancing model based on the SSAWF (Strong Suspend And Weak Forecast) mechanism. This model uses strong suspend mechanism and cubic exponential smoothing prediction method based on AHP algorithm for dynamic load balancing scheduling. Results of the experiments show that the improved model has more positive influence on read/write performance of the cluster under abnormal system transient performance, high concurrency and high system load interaction, that's to say the load balancing effect is better than the traditional load balancing.

KEYWORDS
AHP, Cubic Exponential Smoothing Method, Load-Balancing, SSAWF, Web Server Cluster

1. INTRODUCTION

With the rapid development of computer technology and the Internet, Internet service is not only an important hinge within personal relationship and between human and society, but also the door of breakthrough for all kinds of industries to accelerate their transformation and upgrading. Big data strategy is firstly proposed on World Internet Conference 2015. It indicates that the importance, scale and complexity of internet data will change from quantity to quality. The increasing of volatility of man-machine interaction, as well as the rapid increase of internet data both in scale and importance, makes Web server face a severer challenge. To cope with this challenge, it can build a Web server cluster which is a group of independent computers, connected by network and managed by a single system model (Menzel, Ranjan, Wang, Khan, & Chen, 2014). It can use many servers to execute in parallel and it can establish a quick information communication channel to reduce the pressure on each server of the cluster and improve the whole performance of the cluster. It has high scalability, high availability and high cost performance. Thus, building Web server cluster become the main method at home and abroad (Solar, Suppi, & Luque, 2012).

Taking into account the size, importance (priority), uncertainty of the task, the difference of the performance of each node (processing unit) in Web server cluster and the difference of the
communication overhead, even in the initial stage, tasks are allocated by adopting fair principle, but in actual operation process, each node of load may also gradually appear imbalance (Ok & Park, 2004). This phenomenon makes Web server cluster not take full advantage of system recourses, so it causes the whole performance of the cluster decline (Adler, Chakrabarti, Mitzenmacher, & Rasmussen, 2010). To improve the efficiency of the cluster, the main method is to use load balancing technique (Chen & Li, 2015). Therefore, the research on load balancing has been continued until now, and it will always be the focus of scholars all over the world.

For Web server cluster, load balancing is a classical combinatorial optimization problem. The description of the problem is: Given a number of tasks, to find a method to allocate these tasks to each node in the cluster. This allocation method is to use one or more objective function to optimize the cluster 9 (Grosu & Chronopoulos, 2002). Specifically, the problem of load balancing in Web server cluster is defined as follows:

As shown in Figure 1, $T_1, T_2, \ldots, T_n$ is a set of tasks to be processed, $S_1, S_2, \ldots, S_m$ is each server of Web server cluster. Each server communicate through the network, the task scheduler adopts strategies to allocate $n$ tasks to $m$ servers. Strategies are defined in allocation mechanism $(M)$. In the process of task allocation, it is necessary to consider how to achieve fair allocation of tasks among each node, so that the overhead of whole process is minimized.

Load balance method is mainly divided into static load balancing and dynamic load balancing (Wang, 2013). Static load balancing is based on the stateless, estimated load in the beginning to build a cluster system and determine a scheduling mechanism, it does not consider the actual load in the system operation. This means that the static information of all nodes must be clearly determined before the cluster is built (Effaparvar & Garshab, 2014). Traditional static load balancing methods include Round-Robin (RR), Weighted Round Robin (WRR), Least Connections (LC) and the like. However, the actual operating environment is not always in a stable state, the application will produce real-time load or maybe appear dramatic load changes, and even the unknown behavior (Wei & Schaft, 2003). In this case, the static load balancing method cannot be able to generate an effective load balancing scheduling. Because of unpredictability and interactivity of network environment are enhanced, load value which is estimated before the cluster running is less accurate and static load balancing is not adopted as time goes by. The analysis and scheduling of dynamic load balancing is based on real-time load of the system. In the system operation process, dynamic load balancing allocates the task to many processing units and the number of tasks is adjusted in real time according to load of each processing unit. It order to match the load of each processing unit with own performance as far as possible (Wang, Tao, Qian, & Kuang, 2016). Weighted Least Connections (WLC) is a classic dynamic load balancing algorithm, which is based on the current number of connections $(R_m)$ and

**Figure 1. Description of load balancing in Web server cluster**
14 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the product's webpage:

www.igi-global.com/article/an-improved-dynamic-load-balancing-model/182535?camid=4v1


www.igi-global.com/e-resources/library-recommendation/?id=2

Related Content

Factors Affecting Team Motivation: A Survey of Finnish Software Engineers
www.igi-global.com/article/factors-affecting-team-motivation/126954?camid=4v1a

Combinatorial Testing
www.igi-global.com/chapter/combinatorial-testing/37033?camid=4v1a
Supporting Model-Driven Development: Key Concepts and Support Approaches
Rita Suzana Pitangueira Maciel, Ana Patrícia F. Magalhães Mascarenhas, Ramon Araújo Gomes and João Pedro D. B. de Queiroz (2014). *Handbook of Research on Emerging Advancements and Technologies in Software Engineering* (pp. 176-212). [www.igi-global.com/chapter/supporting-model-driven-development/108617?camid=4v1a](www.igi-global.com/chapter/supporting-model-driven-development/108617?camid=4v1a)

Responding to Ongoing Change: Challenges for Information Systems Modeling