



ELSEVIER

Computer Networks 40 (2002) 235–256

COMPUTER
NETWORKS

www.elsevier.com/locate/comnet

Modeling and performance analysis of QoS-aware load balancing of Web-server clusters

Zhiguang Shan ^a, Chuang Lin ^{a,*}, Dan C. Marinescu ^b, Yang Yang ^c

^a Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

^b School of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32828, USA

^c Information Engineering School, University of Science and Technology Beijing, Beijing 100083, China

Received 30 March 2001; received in revised form 30 November 2001; accepted 1 March 2002

Responsible Editor: I. Niholaidis

Abstract

This paper introduces mechanisms to correlate contents and priorities of incoming HTTP requests used for server process scheduling with the load balancing policies for Web-server clusters. This approach enables both load balancing and Web quality of service (QoS). Another contribution is a modeling and analysis technique based on stochastic high-level Petri net methods for QoS-aware load balancing. We propose an approximate analysis technique to reduce the complexity of the model.

© 2002 Elsevier Science B.V. All rights reserved.

Keywords: Web quality of service; Load balancing; Web-server cluster; Stochastic high-level Petri net; Performance analysis

1. Introduction and motivation

The Internet is evolving from a communication and browsing infrastructure to a medium for conducting business and selling services. Enterprises and service providers are increasingly motivated to migrate mission-critical services to the Web [1]; on-line banking, stock trading, reservations, product merchandising, to name a few, are services being offered via the Web. Since not all

transactions are equally important to the clients or to the servers, e-commerce applications generally desire preferential treatment foreign to the egalitarian philosophy of TCP/IP and HTTP. To that end, Web servers must be capable of providing differentiated quality of service (QoS).

This paper introduces mechanisms to correlate contents and priorities of incoming HTTP requests used for server process scheduling with the load balancing policies for Web-server clusters. This approach enables both load balancing and Web QoS. Another contribution is a modeling and analysis technique based on the stochastic high-level Petri net (SHLPN) [24] to investigate QoS-aware load balancing. We propose an approximate analysis technique based on the decomposition

* Corresponding author. Tel.: +86-1062783596; fax: +86-1062771138.

E-mail addresses: shanzhiguang@263.net (Z. Shan), chlin@tsinghua.edu.cn (C. Lin), dcm@cs.ucf.edu (D.C. Marinescu), yyang@ustb.edu.cn (Y. Yang).

and iteration methods to reduce the complexity of the model.

The paper is organized as follows: In Section 2 we review related work and in Section 3 we discuss scalable server architectures and load sharing models. In Section 4 we provide an informal introduction to SHLPNs and then present the SHLPN model of the Web-server cluster. In Section 5, we specify the server process scheduling policy and load balancing policy concerned, and introduce a QoS-aware load balancing policy based on them. The metrics used in performance analysis is also described. In Section 6 we present the numerical results by two examples to show the performance benefits of QoS-aware load balancing. In Section 7 we propose an approximate analysis technique to cope with the state-space explosion problem, and give the corresponding numerical results and validation. Finally, Section 8 concludes the paper with discussions on the future work.

2. Related work

Bhatti and Friedrich propose an architecture for tiered Web services [2]. Almeida et al. investigate priority-based request scheduling at user and kernel levels [3]. They modify the Apache server and the Linux kernel; the server includes a scheduler process and the kernel maps request priorities into priorities of the HTTP processes handling them.

Pandey et al. present a QoS model for Web servers that enables a site to customize the response to external requests by setting priorities among page requests and allocating server resources [4]. Bhoj et al. implement a QoS-enabled Web server using QoS-aware middleware that implements prioritization of requests, predictive queue control and multi-stage admission control without any modification to the operating system or the Web server software [5].

Crovella et al. [6] propose a policy favoring short connections for static files. The shortest remaining processing time (SRPT) scheduling policy is analyzed in [7,8]. Other mechanisms and policies for Web QoS, such as operating system control [9], server-side application-level-only mechanisms [10], Web content adaptation for server resource man-

agement [11], control-theoretical approach for performance guarantees [12] are discussed. Commercial products such as Hewlett-Packard's Web-QoS [13] and IBM's WebSphere [14] now include some of these QoS capabilities.

Cardellini et al. review the state of the art in load balancing techniques on distributed Web-server systems [15]. Bryhni et al. present a comparison of load balancing methods for scalable Web servers [16]. Schroeder et al. [17] overview the clustering technologies for Web-server clusters. They classify server clusters into three categories based upon the dispatching strategy:

- L4/2—layer four switching with layer two packet forwarding,
- L4/3—layer four switching with layer three packet forwarding, and
- L7—layer seven switching.

3. Scalable Web-server architecture and load sharing models

A recent study of the workload placed on a Web server performed by Arlitt and Jin [18] reports results collected during the 1998 World Soccer Cup held in Paris. According to the study, the Web server got an average of 10,756 requests per minute; the total number of requests over a period of nearly three months was 1,352,804,107. The total amount of data transferred was close to 5 Terabytes and the volume of data stored on the server was 0.3 Gbytes. These figures are typical for high volume servers. Clearly, a single HTTP server cannot possibly sustain such a load thus the need for scalable Web-server architectures. In this section we discuss several approaches to ensure scalability of Web servers, the problem of load balancing among servers in a Web-server cluster, and define the concepts of *End-to-End QoS* and *Web server QoS*.

A first approach to ensure Web-server scalability is to have multiple mirror images of a Web server, each located possibly in a different domain and with its own IP address. Balancing the load among the servers can be achieved using the domain name services (DNS) as shown in Fig. 1.

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات