

一种新的互联网应用体系结构

A New Web Application Architecture

李景峰¹ 李琰² 陈平¹

(西安电子科技大学软件工程研究所¹ 通信工程学院² 西安710071)

Abstract As the increasing popularity and scale of Web application, the architecture of Web application is becoming more and more complex, for which the simple Web application architecture is not suit. In this paper, based on the analysis of complex Web application, we propose a new Web application architecture—"Browser-Web Server-Application Server-Database" for the complex Web applications. It facilitates the rapid development and maintenance of complex Web applications and the reuse of previously gathered experience, and provides a method for the integration of legacy information system with Web.

Keywords Web application, Architecture, Four-layer architecture

1 引言

Internet, Intranet, Extranet 和 WWW 的发展已经对工业、商业、金融业、娱乐业、教育、政府部门以及我们个人的工作和生活产生了巨大的影响。现在许多原有的信息系统和数据库系统正在被移植到 Internet 和互联网环境中,一些新的复杂的分布式系统也出现在 Web 环境中,互联网应用系统变得日益广泛。同时,互联网应用系统的复杂性也随之增加。从简单的用于信息发布的网站到基于互联网的大型企业的管理系统、电子商务系统,互联网应用系统的规模越来越大,结构变得越来越复杂^[1]。这样,现有的互联网应用体系结构已不能适应复杂的互联网应用系统。

本文针对该问题,分析了现有的互联网应用体系结构的不足,结合复杂互联网应用系统的特点,提出了一种新的互联网应用体系结构,并将其与现有的互联网应用体系结构进行了比较。

2 现有的互联网应用体系结构

互联网应用系统是指依赖于互联网环境的应用系统^[2]。小到由几个页面构成的个人网站,大到由数万页构成的大型 ISP/ICP 网站,基于互联网的企业管理系统、电子商务系统等都属于互联网应用系统。互联网应用系统都采用 Browser/Server (B/S) 结构。但随着互联网技术的发展和用户对动态性、时效性的需求,互联网应用的体系结构也在不断变化。

2.1 互联网应用的 B/S 结构

早期的互联网应用系统就是以信息发布为主要目的的网站^[3]。它通常由静态的网页通过超链接组织而

成,用户只能通过导航与系统进行交互。它的体系结构是典型的 B/S 结构,如图1所示。用户在浏览器中输入 URL,浏览器用超文本传输协议(HTTP)向 Web 服务器请求一个 HTML 文件,Web 服务器接收浏览器的请求,向浏览器发送此页。

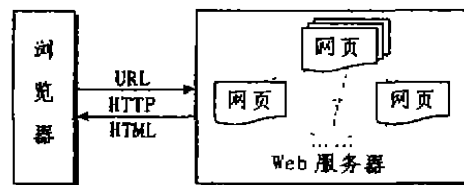


图1 互联网应用的 B/S 结构

该结构虽然简单,但它存在着不足之处。在这种结构下,网站上页面的信息是静态的,无法根据用户的需求和实际情况作出相应的变化。当浏览器通过 HTTP 协议向站点的 Web 服务器申请页面时,Web 服务器就会将已设计好的静态的 HTML 文件传送给浏览器。若要更新页面的内容,只能用非在线的手动方式更新 HTML 文件。这种结构只给用户和浏览器之间提供有限的交互,而且当网站规模比较大时,对它的维护将成为一个难题^[4]。

2.2 互联网应用的三层结构

随着互联网应用对动态性、时效性、灵活性的需求,出现了以数据库为中心的互联网应用系统。在这种系统中,Web 服务器方不仅仅是静态的网页,还包括一些访问数据库的机制(如 CGI 程序、ASP 程序、IS-API 动态连接库、Java Servlet 等)。当用户通过浏览器提交 HTTP 请求后,Web 服务器执行一个特定的应用

程序,而不是一个简单的 HTML 文件。该应用程序分析输入的数据,根据不同的数据内容将相应的执行结果(通常是数据库查询的结果集)以 HTML 的格式传送给浏览器,数据库的数据可以随时变化,而服务器上执行的应用程序却不必改变,客户端始终会得到最新的网页。这种互联网应用系统采用的就是“浏览器-Web 服务器-数据库”的三层结构,如图2所示。

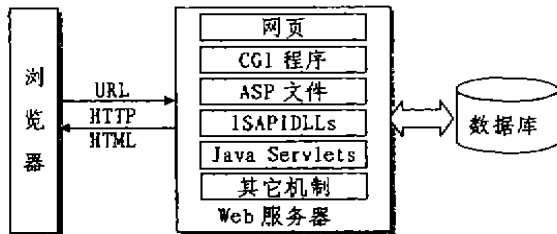


图2 互联网应用的三层结构

这种结构是目前互联网应用系统采用最多的体系结构。采用这种结构,Web 服务器通过发送给用户数据库中存储的最新的消息,来保证互联网应用系统的动态性和实时性。由于一些网页是动态生成的,系统维护任务也相对减少。同时用户与系统的交互性也增强了。

但是,随着互联网应用渗透到各个领域以及其规模的增大和复杂性的增加,互联网应用系统的应用逻辑已经成为系统的核心。在互联网应用的三层结构中,系统的应用逻辑是 Web 服务器运行相应程序时被隐式执行,这种方式对于以数据库为中心的互联网应用是可以的,但对于以应用逻辑为中心的互联网应用它就不适合,因为后者的应用逻辑一般比较复杂,它需要专门的应用程序来协作完成,所以需要采用新的体系结构来支持新型的互联网应用。

3 互联网应用的四层结构

3.1 复杂互联网应用系统的特点

目前,互联网应用系统的规模越来越大,结构越来越复杂,出现了网上书店、网上拍卖、电子商务等复杂的互联网应用系统。在这些系统中,虽然也需要数据库,但它已经不是系统的核心,系统的核心在于系统应用逻辑的执行。通过对这类互联网应用系统的分析,得出它们有以下特点:

- 系统的应用逻辑比较复杂,对用户输入的处理通常需要一个完整的工作流程,应用逻辑相对稳定;
- 系统更具开放性,通常需要与原有的信息系统和数据库系统进行交互;
- 系统具有实时事务处理的能力。

3.2 互联网应用的四层结构

对于复杂的互联网应用系统,将系统功能隐含在其它程序里不仅概念上不清晰,而且设计时容易产生混乱。所以,三层结构已不适应,需要有单独的一层来执行系统的应用逻辑。这样,我们可以在 Web 服务器和数据库中间增加一个应用服务器层,由它来完成系统的功能,从而,我们得到了“浏览器-Web 服务器-应用服务器-数据库”互联网应用的四层结构,如图3所示。

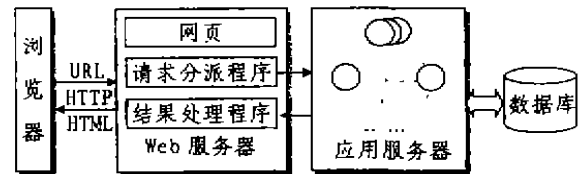


图3 互联网应用的四层结构

在该结构中,Web 服务器层主要负责接收请求、转换请求格式、将转换后的请求分发给应用服务器以及接收应用服务器发送来的处理结果,并将其转换为 HTML(或 XML)文件,应用服务器层主要是为具体应用提供必要的系统支持,即通过应用服务器上程序的运行来完成应用系统的功能(在程序运行过程中应用服务器通常要访问数据库)。应用服务器上的程序可以由可重用构件或者是软件功能包搭建而成。

采用这种结构,Web 服务器接收到用户通过浏览器提交的 HTTP 请求后,并不进行处理,而是作相应的格式转换,再转发给应用服务器。应用服务器对请求进行处理并将处理的结果返回给 Web 服务器。Web 服务器再将执行的结果转换成 HTML(或 XML)文件传送给浏览器。在整个过程中,系统的应用逻辑是在应用服务器层被执行的。采用互联网应用的四层结构有以下优点:

- 便于服务器方软件的重用。目前,对互联网应用系统的重用主要集中在代码重用上^[5]。而在四层结构中,由于互联网应用系统的逻辑完全在应用服务器上执行,因此针对该类型的互联网应用系统,其分析、设计以及实现代码都可以被充分地重用。
- 便于以应用逻辑为中心的互联网应用系统的快速开发。由于系统被分层,可以针对不同的层采用不同的软件开发技术,对 Web 服务器层,我们使用互联网特有的开发技术和工具,而对于应用服务器层我们可以使用传统的软件工程方法,而且只要规定好这两层之间的交互协议,它们的开发可以同时进行。
- 便于系统功能的维护。对于互联网应用系统,其功能的维护通常是个难题^[4]。采用四层结构,可以使得系统功能的维护变得比较容易。当系统的功能发生改变时,不必改变 Web 服务器方的代码,只要保持应用

服务器对外接口不变,就能顺利地调整系统的功能,而该过程对用户是透明的。

·为传统的信息系统移植到互联网环境中提供了一种办法。现在,许多原有的信息系统将要移植到互联网环境中。若采用三层结构,我们不仅要完成网站页面的设计,还要重新编写大量的程序以实现原有信息系统的应用逻辑。这样不仅做了大量的重复劳动,而且其可靠性也没有保障。采用四层结构,只需在原有信息系统服务器程序上“粘”一块程序,相应地在 Web 服务器方也“粘”一块程序,就可以很快地将原有系统移植到互联网环境中。

当然,互联网应用的四层结构也有不足之处,由于用户的请求被转发了一次,这可能会影响系统运行的效率。但任何事情都具有两面性,笔者认为采用四层结构的好处:可重用性、易开发性、易维护性、易集成性,远远大于其带来的负面影响。另外,必须明确一点,四层结构并不是对于任何互联网应用系统都是适合的。只有对于那些复杂的以应用逻辑为中心的互联网应用系统,四层结构才是比较适合的。

3.3 几种结构之间的比较

下面我们从适用对象、交互性、动态性、性能、易维护性和可重用性几方面对典型的 B/S 结构、三层结构和四层结构做一个简单比较(表1)。

结束语 随着互联网应用的日益广泛,出现了以应用逻辑为中心的复杂的互联网应用系统,原有的支持以数据库为中心的互联网应用的体系结构已经不适合,必须采用新的体系结构来支持复杂的互联网应用系统。本文在分析复杂互联网应用的基础上,提出了一种针对复杂互联网应用系统的新的互联网应用体系结

构——“浏览器-Web 服务器-应用服务器-数据库”。该结构有利于复杂互联网应用系统的快速开发、维护和重用,也为原有信息系统移植到互联网环境中提供了一个很好的办法。

表1

	适用对象	交互性	动态性	性能	易维护性	可重用性
典型的 B/S 结构	以信息发布为主的静态的网站	差	差	好	差	差
三层结构	以数据库为中心的互联网应用	较好	好	较好	中等	中等
四层结构	以应用逻辑为中心的互联网应用	好	好	稍差	好	好

参考文献

(上接第100页)

- Akl S.G. The Design and Analysis of Parallel Algorithms. Prentice-Hall, Englewood Cliffs, NJ, 1989
- Chellappa, R, Chatterjee S. Classification of Textures Using Gaussian Markov Random Fields. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1985, 33(August): 959~963
- Cohen F S, Cooper D B. Simple parallel hierarchical and relaxation algorithms for segmenting noncausal Markovian fields. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-9(March): 195~219
- JaJa J. An Introduction to Parallel Algorithms. Addison-Wesley Publishing Company, New York, 1992
- Bader D A, JaJa J, Chellappa R. Scalable Data Parallel Algorithms for Texture Synthesis and Compression Using Gibbs Random Fields. [Technical Report CS-TR-3123 and

- Gellersen H, Gaedke M. Object-Oriented Web Application Development. Internet Computing, 1999 (January-February): 60~68
- Gellersen H W, Wiche R, Gaedke M. WebComposition: An Object-Oriented Support System for the Web Engineering Lifecycle. 1996. Available at: <http://proceedings.www6conf.org/HyperNews/get/PAPER232.html>
- 熊忠阳, 张玉芳, 吴中福. 三层结构中的数据库访问技术. 计算机科学, 2000, 27(4): 95~97
- Fan Xin, Chen Jian. Design for Maintenance: Experience of Developing an ICP Web Site. In: 1st ICSE Workshop on Web Engineering, Los Angeles, USA, 1999
- Zhao Wei-quan, Chen Jian. CoOWA: A Component Oriented Web Application Model. In: Proc. of 34th Intl. Conf. on TOOLS Asia. IEEE Computer Society, Nanjing, 1999

UMIACS-TR-93-80]. UMIACS and Electrical Engineering, University of Maryland, College Park, MD, August 1993

- Johnsson S L, Jacquemin M, Krawitz R L. Communications Efficient Multi-Processor FFT. Journal of Computational Physics, 1992, 102: 381~397
- Palmer J, Steele Jr C L. Connection Machine Model CM-5 System Overview. In The Fourth Symposium on the Frontiers of Massively Parallel Computation, IEEE Computer Society Press, Los Alamitos, CA, 1992. 474~483
- Lu H, Dwarkadas S, Cox A, Zwaenepoel W. Quantifying the Performance Differences Between PVM and TreadMarks. Journal of Parallel and Distributed Computing, 1997(42)2: 65~78
- Castleman K R. Digital Image Processing. Prentice Hall, 1997