FUWUQIDUAN RUANJIAN XINGNENG FENXI HE ZHEN DUAN

服务器端软件性能分析和诊断

主　编　刘雪梅
副主编　柳永坡

北京邮电大学出版社
www.buptpress.com

服务器端软件性能分析和诊断

主　编　刘雪梅
副主编　柳永坡

北京邮电大学出版社
www. buptpress. com

内 容 简 介

本书以软件性能分析和诊断为研究目标，以 Web 服务器为应用背景，对 Web 服务器端软件性能分析和诊断方法进行了深入的研究。具体的研究内容包括构建了 Web 应用性能的网状度量模型；创新性地提出了性能问题模式的概念；构建了性能问题的诊断模型。本书对 Web 服务器端软件性能的度量和分析及诊断都进行了建模和分析，并通过实际的实验数据加以验证，旨在帮助读者彻底理解服务器端软件性能测试的原理和本质。本书结尾，作者精心总结了 Web 服务器端软件性能度量和分析的实验结论，颇具参考价值。

本书适合初入软件测试行业的研发人员、技术工程师阅读，同时适合资深软件测试行业人士用以互相切磋交流提高。另外，网络工程师、网管、服务器软硬件开发与 Web 开发者、数据库开发者以及相关专业师生等也非常适合阅读本书。

图书在版编目(CIP)数据

服务器端软件性能分析和诊断/刘雪梅主编．--北京：北京邮电大学出版社，2011.7
ISBN 978-7-5635-2648-2

Ⅰ.①服… Ⅱ.①刘… Ⅲ.①服务器软件—性能分析②服务器软件—诊断技术Ⅳ.①TP393.09

中国版本图书馆 CIP 数据核字(2011)第 120195 号

书　　名：服务器端软件性能分析和诊断
主　　编：刘雪梅
责任编辑：何芯逸
出版发行：北京邮电大学出版社
社　　址：北京市海淀区西土城路 10 号(邮编：100876)
发 行 部：电话：010-62282185　传真：010-62283578
E-mail：publish@bupt.edu.cn
经　　销：各地新华书店
印　　刷：北京源海印刷有限责任公司
开　　本：787 mm×960 mm　1/16
印　　张：9.75
字　　数：209 千字
版　　次：2011 年 7 月第 1 版　2011 年 7 月第 1 次印刷

ISBN 978-7-5635-2648-2　　定　价：20.00 元

前　言

随着 Internet 的迅速发展，基于 Web 的应用越来越深入人们的工作和生活，许多传统的信息系统被移植到互联网上，电子商务等新的应用模式也在不断涌现，Web 正以其广泛性、交互性、快捷性和易用性等特点受到越来越多的企业和个人的青睐。性能是 Web 应用成功的一个重要因素，减少 Web 服务器端软件的缺陷，保证和提高 Web 服务器软件的质量已成为软件测试研究的重要内容。

目前针对 Web 服务器端软件进行性能测试的相关方法和工具比较成熟，应用也比较广泛。但是，在发现性能问题之后，却没有较好的方法和工具能够支持性能问题的进一步诊断；或者诊断结果单一，不能支持多个性能问题的联合诊断。因此，本书以软件性能分析和诊断为研究目标，以 Web 服务器为应用背景，对 Web 服务器端软件性能分析和诊断方法进行了深入的研究。即通过对 Web 日志进行数据挖掘，获取性能度量指标，定义性能问题模式。作为性能分析和诊断的基础，所提出的性能分析和诊断方法可以准确发现系统中存在的性能问题，并能确切定位部分性能问题的位置，提高了性能分析和诊断的自动化程度。

本书共分 5 章，内容包括 Web 服务器端软件性能度量和诊断的研究背景、Web 服务器端软件性能测试的研究现状、性能度量模型、性能缺陷的表征模式、性能诊断方法等。

本书由刘雪梅任主编，柳永坡任副主编。在编写过程中，得到了北京航空航天大学软件所、北京工业大学软件学院、哈尔滨工程大学计算机学院的大力支持，在此表示衷心的感谢！由于作者水平有限，书中纰漏在所难免，恳请广大读者批评指正。

编　者

目　录

第1章 绪 论

互联网的应用普及只用了短短十几年的时间，从最初的静态信息浏览快速发展到了动态信息搜索和多媒体应用。随着网络软硬件技术的巨大进步，大多数的数字信息化应用可以通过网络模式来实现。

中国互联网络信息中心在《第27次中国互联网络发展状况统计报告》中的数据显示，截至2010年年底，我国网民数达到4.57亿，较2009年增加7330万人，互联网普及率攀升至34.3%。图1-1给出自2003年以来，中国网民的规模与普及率。

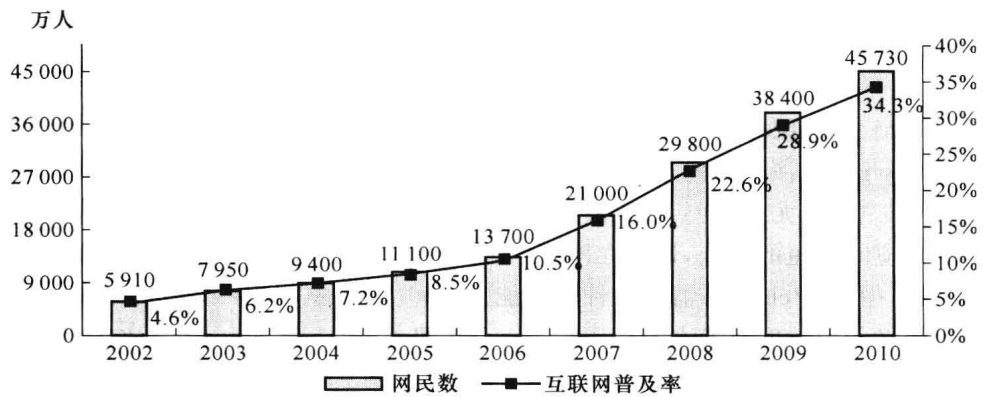

图1-1 中国网民规模与普及率

互联网正在改变着人们的工作、生活和学习，人们可以在互联网上建立一个虚拟的电子世界，在这个虚拟的世界里，人们的思想和概念可以在几分钟内传遍全世界。全世界的商家们拥有了一个更为灵活和快速的媒体，通过它可以与它们自己的员工、潜在的客户乃至世界上任何一个人沟通，电子商务的概念也随之而来。借助于互联网，通过动态的交互式信息发布，诸如网上购物、网上银行、网上书店等一系列在线电子商务服务系统也越来越盛行。表1-1列出了2007—2010年电子商务类应用用户的发展情况。

表 1-1　2007—2010 年电子商务类应用用户对比

	2007 年底		2010 年底		变化	
	使用率	网民规模/万人	使用率	网民规模/万人	增长量/万人	增长率
网络购物	22.1%	4 600	70.5%	11 500	6 900	70.3%
网络售物	—	—	5.9%	4 200	—	—
网上支付	15.8%	3 300	30.6%	8 200	4 900	60.4%
旅行预订	—	—	20.6%	4 900	—	—

1.1　研究背景及意义

World Wide Web 简称 Web 或 WWW，中文名字万维网，是自 20 世纪 90 年代以来最重要的 Internet 应用。本质上 Web 应用就是 client-server 应用，即客户端是 Web 浏览器，服务器端是 Web 或者应用服务器。作为一种资源的组织和表达机制，Web 已成为 Internet 最主要的信息传送媒介。

基于 Web 的各种 B/S 模式网络应用逐步深入人们的工作和生活，越来越多的公司通过 Web 服务器与客户进行业务交流。通过网络获取信息是以前大多数网民上网的主要目的，随着软件系统的设计体系向着基于 Web 的分布式系统发展[1]，获取信息已经不是问题，人们开始关注网络体验，因此如何迅速响应用户的访问是优秀网站的一个重要标志，对商业网站来说尤为重要。显然，响应时间长的 Web 应用会使用户放弃其站点，而转向其他的响应时间短的站点，这对在网上销售商品的商家来说无疑会减少潜在客户，造成很大的损失，因此 Web 服务器端软件的性能是 Web 应用成功的一个重要因素。

负载测试领域首席技术专家 Alberto Savoia 早在 2001 年指出，当站点的性能缺陷达到无法忍受的程度时，就会导致用户过早终止在该网站的事务[2]。研究表明，每年由于不能接受的下载速度导致的损失达十几亿美元。据 Zona 研究报告统计[3]：下载时间每减少 1 s，用户放弃率就会下降 6%～8%；全球商务网站每年因性能缺陷造成损失 43.5 亿美元，占总损失的 15%[4]。

显然，一个性能低下、频繁出现异常事件的 Web 应用无法让客户满意，只会让企业失去客户，因此如何提高和保证 Web 应用的性能已经成为很多企业棘手的问题。一般而言，导致 Web 应用性能缺陷的原因主要有两点：

① Web 服务器端软件自身的缺陷；

② Web 服务器部署环境的配置。

显然第①点是 Web 应用性能产生问题的主要原因，甚至是系统性能瓶颈所在[5]。因此，目前在软件测试领域侧重于研究 Web 应用软件自身的缺陷，以提高性能、保证软件质

量。人们研究了提高软件性能的工作量分布，发现其中大约75%的工作量都用于性能检测、分析和诊断问题，只有25%用于修复问题[6]，这充分说明了Web服务器端软件的性能分析和诊断对于提高Web应用性能的重要意义。

在有关服务器端软件性能缺陷的研究中，性能测试的方法和工具已比较成熟，应用较广泛，如虚拟用户方法[12]、WUS方法[13]、对象驱动方法[14]、JMeter测试工具[35]、LoadRunner测试工具[36]、OpenSTA测试工具[37]，但这些方法和工具在发现相关性能缺陷后却不能对性能缺陷进行分析和诊断。

服务器端软件的性能度量和分析方法可以借鉴并行程序性能研究的部分成果，如基于程序执行状态和执行轨迹分析[38-46]、内存使用问题分析[47-52]，但这些方法针对的并行程序规模较小，不能适应Web应用中的大规模并发的持久化对象、海量的执行轨迹数据，且分析结果单一，不能支持多个性能缺陷的联合诊断。

针对Web服务器的性能，目前多采用日志挖掘分析的方法，从统计角度分析用户浏览或访问模式、入侵模式[53-59]，进而设计海量日志分析算法用于性能分析[60-67]，但这些研究对性能分析和诊断提供的支持非常有限，效果也不太好，更重要的是分析结果难以重用。目前故障定位研究主要针对测试中所发现的功能性问题或失效[68-78]，且分析的数据量一般都不大，没有考虑海量数据的处理，这些研究成果都难以直接用于性能缺陷的诊断和定位。

因此，开展Web服务器端软件的性能分析和诊断研究对于切实解决Web应用的性能缺陷，是一项重要而紧迫的任务。本书从Web应用性能缺陷本身出发，通过挖掘Web日志数据，采用分类和聚类方式来分析影响各类性能缺陷的性能指标，找到性能缺陷发生的原因，进而发现性能缺陷的影响因素，并对这些影响因素进一步挖掘和分析后，来定义Web应用的性能缺陷表征模式，最终基于性能缺陷表征模式建立一套更加科学有效的诊断方法。该研究成果必将具有广阔的市场应用前景。

1.2 Web服务器端软件性能测试概述

Web应用一般需要长时间连续运行，且无人值守。因此，功能测试之后就会把Web应用部署在目标环境中进行性能测试，如压力测试、负载测试、鲁棒性测试等。性能是指计算机总体的工作效率，包括响应时间、吞吐量及可用性等。响应时间是指从用户发出请求到得到响应的整个过程的时间；吞吐量是指在给定的时间内系统处理的请求数；可用性是指系统能够正常工作的时间比例。因此，性能测试的目的体现在5个方面[11]：识别系统的弱点、评估系统的能力、检测软件的问题、系统调优、验证稳定性和可靠性。

对于Web应用，性能测试是其测试过程的一个重要组成部分。性能测试主要用来识别系统性能瓶颈，为将来的测试工作建立基线(baseline)，支持性能调优(tuning)，确定符合性能目标和需求，并且/或者搜集其他的性能相关数据为被测应用的综合质量决策提供

信息。另外，性能测试和分析的结果可用来评估硬件配置是否满足应用系统较好的工作。

性能度量是通过定义一系列可以反映程序性能的指标，并从程序实际运行的数据中获得度量结果的过程。度量性能的过程一般使用统计证据对目标进行详细定义。

性能管理就是一些用来保证系统始终有效和高效工作的活动。主要涉及组织、学科、产品或者服务的建立过程以及使用的性能等。应用性能管理，属于系统管理学科范围，集中监测和管理软件应用的服务可用性。性能管理可以看做是使用 IT 工具检测、诊断、修正并报告应用性能满足/超出终端用户和商业的期望的过程。

性能调优指通过调整应用的配置、算法策略、数据缓存策略等以期改善系统性能的过程，目的是提高系统性能。大部分系统都会在增加负载时造成性能一定程度的下降，性能调优是提高系统可扩展性的重要手段。性能调优主要包括以下一些常见的技术：

性能分析，一般也叫做剖析，是指使用程序运行时收集到的信息来研究程序的行为，目的是确定对程序的哪个部分进行优化，profiler 是一个依据程序执行行为，尤其是函数调用的频率和持续时间的性能分析工具；

性能工程，是一个包含任务、技能、活动、时间、工具以及可交付使用等满足系统非功能需求的学科，如增加商业税收，减少系统失效，项目延期以及避免不必要的资源使用和工作；

代码优化，是通过重写程序的特殊部分来提高性能的方法，一般是指对算法的改进或使用更好的算法；

负载均衡，是基于分布式系统潜在的候选机器的繁忙程度来分配操作的执行以获取更高的效率。

本书中的性能调优主要采用了性能分析方法。

Web 技术在近年来得到了飞速的发展，从原来单纯的 HTML、CGI，到后来的 JavaScript、Java Applet、ActiveX 控件，再到 ASP、JSP、PHP 等脚本技术，直到现在最新的 J2EE 架构、.NET 架构、Web 服务技术等，Web 应用的开发受到了前所未有的重视，基于 Web 的分布式系统已逐步成为软件系统设计体系的主要发展方向。图 1-2 给出了 Web 应用的体系结构。

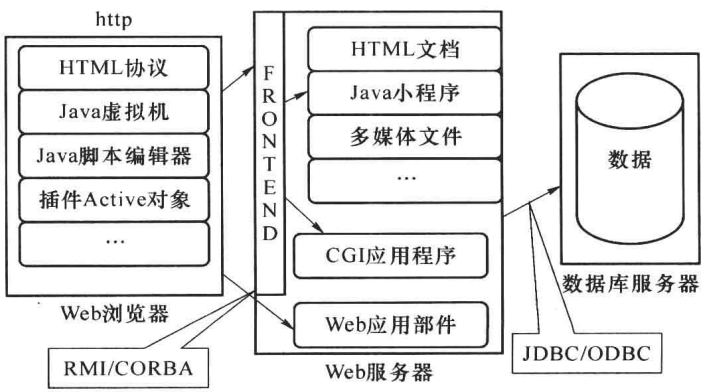

图 1-2　Web 应用体系结构

Web 应用系统之所以开发效率高、易维护、易扩展，主要得益于组件化的分层体系结构，然而，这些优点也会给 Web 应用系统自身带来潜在的性能缺陷，成为导致性能瓶颈的重要原因，Web 应用软件的特殊性使得性能测试也更加困难，其特殊性如表 1-2[1] 所示。

表 1-2　Web 应用软件的特殊性

序号	特征体现	描　述
1	体系结构	多层架构，划分为：表示层、业务逻辑层、数据层
2	实现技术	使用了 XML、HTML、JavaScript、VBScript、Database、PHP、ASP、JSP、CGI 等多种编程技术，导致系统实现复杂
3	组成成分	由 XML、HTML、VBScript、JavaScript、Form、ASP、ISAPI、CGI、JSP、Web services、JavaBeans 等构成，成分繁多
4	运行机制	具有分布式、并发、动态、实时交互的特点
5	运行过程	不确定性
6	运行环境	Web 应用性能与环境及负载有关
7	设计开发	需求不明确，设计开发时间短、变化快

Web 应用系统的特殊性使得测试比普通程序复杂得多，传统的测试技术无法适用，必须设计独特的测试体系来支持其性能测试。

Web 应用类型不同，性能测试的方法也不同，目前主要有 3 种方法：虚拟用户方法[12]、WUS 方法[13]、对象驱动方法[14]。本书综合了虚拟用户方法和 WUS 方法的优点，在相关的性能测试实验中采用了上述两种方法来获得实验数据。

对 Web 应用系统进行性能测试时，可以从两方面进行：系统运行平台的性能测试，主要针对客户端、服务器端、网络方面的影响；Web 应用运行性能测试，主要针对工作负载的类型及数量方面的影响。

为了实现系统运行平台的性能测试，可以采用的 Web 应用基准测试有：SPECWeb99[15]、TPC_W[16]、TPC_C[17]、SPECMail2000 等标准。如：文献[18]使用 SPECWeb99 基准对一个实验支撑平台进行了性能测试；文献[19]使用 TPC2C、TPC2W、SPECWeb99 3 个基准对一个模拟实验支撑平台进行了性能测试；文献[20-23]分别对电子商务等不同的支撑平台进行了基准测试。

为了实现 Web 应用运行的性能测试，Web 应用负载的研究成为关键，重点在于如何真实刻画 Web 应用软件的负载特性，形成合理的负载测试模型。本书中进行的就是 Web 应用运行的性能测试。

早期刻画负载特性的方法是通过每秒点击数、每秒访问页面数、每秒访问数等技术指标来刻画，但并不能真实准确刻画负载特性。因此，目前主要采用基于捕捉用户行为、基于文件列表、基于数学分布模型等模拟用户行为的方法[24-26]来刻画。但这种方法具有随

机性、主观性、不充分性的缺点，不能形成有效的负载模拟。因此文献[27-30] 基于 Web 应用服务器的日志文件得到用户行为模型图，来构建负载模型；文献[31]基于充分的日志数据，通过精简用户会话数据集来降低测试代价；文献[32] 在没有充分日志数据的情况下，研究了实现有效负载模型的策略。但因精确刻画负载模型的复杂性，此方向的研究仍将是个难题。

性能度量就是通过定义一系列可以反映程序性能的指标，并从程序实际运行的数据中获得度量结果的过程[33]。性能度量的结果可以用于评估系统的性能好坏、分析性能瓶颈和改进系统性能等。针对 Web 应用性能，有两种常用度量指标：度量资源使用情况，度量响应时间。目前研究多侧重响应时间。

一般 Web 请求的处理过程分为 4 个步骤：

① Web 应用接受来自用户的请求；

② Web 应用把请求转发给后端的 Web 服务；

③ Web 服务处理完毕之后，把结果返回给 Web 应用；

④ Web 应用把最终结果返回给用户。

图 1-3 表示一个 Web 请求在完整的处理过程中不同阶段的响应时间分解。

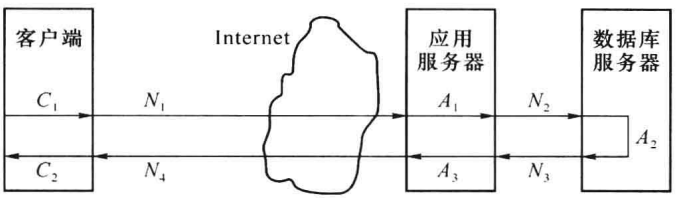

图 1-3　Web 应用响应时间分解[34]

显然，为了方便确定 Web 应用的性能瓶颈，可以把 Web 应用的响应时间分为用户浏览器的处理时间、网络传输时间和服务器的响应时间，即总的响应时间 $=(C_1+C_2)+(N_1+N_2+N_3+N_4)+(A_1+A_2+A_3)$。其中 C_x 表示浏览器处理时间、N_x 表示网络等待时间、A_x 表示应用等待时间。

由于浏览器处理时间和网络传输时间与 Web 应用的业务逻辑并无直接关系，因此本书不对二者进行分析，只研究服务器端的 Web 应用性能缺陷。

Web 应用具有长时间运行和无人值守的特点，一旦系统出现性能缺陷，距离上一次系统维护的时间往往比较长，此时管理员首先需要了解系统性能在这段时间内的变化特征，并根据该系统的具体业务逻辑、用户行为等分析性能瓶颈和缺陷。面对这样的问题，系统管理员所能获得的信息主要包括两类：

① 记录在日志中的关于用户行为、系统响应等信息；

② 管理员上次所作的关于服务器、相关资源、访问策略等的配置信息。

日志数据容易获得，但由于数据量往往非常大，且相关信息分散在多个日志中，因此

难以分析和抽取所需的信息。配置信息也容易获得，但是每个配置对于系统性能的具体影响则难以分析。

一般情况下，Web 应用的性能 P 可表示为如下函数：

$$P(T)=f(B(T),C) \tag{1-1}$$

其中，C 为 Web 应用服务器的配置；T 为满足系统性能目标的连续运行时间；$B(T)$ 为用户行为，可独立变化。在给定的 C 情况下，$P(T)$ 唯一由 $B(T)$ 决定。

实际上，$B(T)$ 为时间序列数据，在时间维上具有叠加性和可分解性，因此可以识别出每个用户（通过 IP 地址）的行为。由于系统具有状态保持能力，因此式(1-1)可进一步表达为

$$P(t_i)=f_C(B(t_i),t_i) \tag{1-2}$$

由于 C 在 T 时间内的常量化，可以把 f 写成 f_C，并把 T 时间分割成 $t_1,\cdots,t_n$ 个小时间段。则 t_i 时间段的系统性能就是 t_i 时间段用户行为和 t_i 时间段内的系统状态的综合函数。面对已经出现的性能缺陷，系统管理员的任务就是通过日志挖掘出 $B(t_i)$ 和 $P(t_i)$，从而分析和构造出相应的函数 f_C，为性能诊断提供直接的支持。当然实践中，管理员不可能去做繁杂的数学推导，而是要对用户行为和系统性能表现进行综合分析，从而发现系统性能缺陷的真正原因。

1976 年，Boehm 等人提出了定量的评价软件质量的模型[127]。他们把软件产品的质量分为 3 个方面：可移植性、可使用性、可维护性，从而得到软件质量的整体评价。1978 年，McCall 等人提出了 3 层次的质量度量模型：质量要素—评价准则—度量[128]。他们把模型的适用阶段分为 3 个：产品运行、产品修正和产品转移，并定义了 11 个质量要素和 23 个评价准则来分别对这些质量要素进行描述，从而反映产品的质量。这两种模型都是网状结构模型，都是从产品的角度进行度量，没有从用户需求的角度进行考虑。而且复杂的网状结构在度量的过程中并不易实现定量化。新版的 ISO/IEC 9126 从用户的角度出发考虑了软件的 6 个质量特性，并规定了 21 个质量子特性。软件质量的度量模型[129]为树形结构，分为 3 层：质量特性—质量子特性—度量，但并没有提到具体的度量方法。

本书的研究目标就是通过挖掘 Web 日志记录来发现用户访问 Web 页面的模式，收集、分析 Web 应用运行时产生的多层次、大量的日志数据，提取用户行为和系统的性能指标，为系统管理员提供分析性能变化曲线规律、不同性能指标之间关系、性能指标的变化与用户行为及其变化之间关系的分析方法和技术手段，同时从用户需求的角度构建性能度量模型，并提供量化的度量方法。

性能评价通过研究负载、配置和性能指标之间的关系，来了解服务器的各项 Web 指标及服务器在高密度大用户使用情况下的表现。性能评价可采用测量和模型两种方法[79,80]，但测量方法只能用于已存在并运行的系统，而且关键在于测量方案和测量手段，因此比较费时。

模型方法就是对评价系统建立模型，求出性能指标，以便对系统进行性能评价。该方

法可以应用于尚未存在的系统，并且测量的工作量较小。但模型方法一般包括许多参数，而这些参数的确定一般需要依据测量结果或者对系统参数的估计，因此要把测量方法与模型方法有机结合，才能正确评价 Web 服务器的性能。

1.3 研究内容

本书的研究重点是 Web 服务器端软件性能分析和诊断方法。为了方便描述所研究的问题及采用的关键技术和方法，首先介绍几个相关概念。

1.3.1 基本概念

1. 软件性能

软件性能一般指一个软件系统（或组件）正确提供其服务的能力和效率，是对软件在确定的平台和配置下对用户请求的响应效率的度量，可以从响应时间、吞吐量、资源利用率及可用性等方面进行度量。

响应时间是指从用户发出请求到得到响应的整个过程所用的时间；吞吐量是指在给定的时间内软件所能正确处理的用户请求数；资源利用率是指系统不同资源的使用状况，比如服务器的 CPU、内存、网络流量等；可用性是指系统能够正常工作的时间比例。软件性能主要取决于运行速度的快慢和需要消耗的系统资源（系统资源中最重要的是内存和 CPU）的多少这两个因素，运行速度太慢的程序将会阻碍系统运行更多的任务。

2. Web 应用性能

Web 应用性能是指 Web 应用作为一类软件提供服务的能力和效率，是 Web 应用在一定的运行环境和用户访问的情况下，能够正确响应用户请求，并随着运行时间的增加和用户数的增长可以保持其响应时间的能力。

Web 应用是一种 B/S 结构的软件，Server 端一般包括 Web 服务器、应用服务器、数据库服务器以及包括操作系统在内的软硬件配置，运行平台及其配置策略会直接影响 Web 应用的性能。吞吐量本身具有一个极限水平，在用户数较少的情况下，随着用户的增多，吞吐量也会增加；在吞吐量饱和的情况下，用户数的增加并不能导致吞吐量增加（即系统在单位时间内已不能处理更多的请求），用户请求的响应时间反而会显著增加。因此多用户情况下，Web 应用性能受到用户访问行为的影响，导致其性能变化具有一定的不确定性。因此，如何发现运行环境各因素与用户访问行为之间的关联关系，对于综合分析与诊断 Web 应用性能具有重要的意义。

针对 Web 应用性能，目前的研究多侧重于响应时间。一般而言的性能多指对于单个用户的请求，系统端到端的响应时间；在多用户并发请求的情况下，使用吞吐量度量来分析系统的性能，即随着并发用户数的增多，系统保持其平均响应时间在可接受范围内的能力。

3. Web 应用性能度量

Web 应用性能度量是一个定义相关性能指标、收集相关数据、指标计算和分析的过程，性能度量的结果可以用于评估系统的性能好坏、分析性能瓶颈和改进系统性能等。

Web 应用性能指标是 Web 应用性能在某个方面的观察，通常具有明确的物理含义和量纲。通过性能指标可以对 Web 应用性能在相关方面进行比较和分析。常见的 Web 应用性能指标包括：响应时间、吞吐量、并发用户数和资源利用率等。

Web 应用性能评估是针对选定的 Web 性能指标，通过进行性能度量，并根据给定的关于性能指标的判定准则对 Web 应用性能进行好坏程度判断的方法。性能指标的判定准则必须符合指标的量纲所规定的计算。一般有两种应用性能的评估方法：度量资源使用情况、度量响应时间。

4. Web 应用响应时间

客户端响应时间指从客户端向服务器发出一个请求到接收到该请求的响应所经历的时间延迟，通常以时间单位来衡量，如秒或毫秒。它是软件性能的一个重要指标，和并发用户数、系统资源利用率等密切相关。客户端响应时间包括服务器端的响应时间和网络传输时间，由于每个客户端网络状况的差异较大，因此本书关注服务器端响应时间，即 Web 应用的服务器端响应时间，度量从用户的一个请求被 Web 服务器受理开始直到该请求被处理完返回响应为止的一段时间。

5. Web 应用性能缺陷

Web 应用性能缺陷是指 Web 应用在运行过程中，针对系统设计所依赖的运行环境和用户请求特征，平均服务器端响应时间过长甚至不响应，或者返回错误的响应。所谓平均服务器端响应时间是指针对多个用户请求，服务器端响应时间的平均值。

性能缺陷不同于功能缺陷，应用系统的功能一般以是否正确实现需求规格为判断标准，而性能缺陷与应用系统的性能设计策略以及运行环境有着密切的关系，很难用正确与否来衡量。性能的判断标准具有相对性，针对不同的运行环境、不同的用户特征、不同的应用场景，性能好坏的判断准则往往不同。

一般来说，Web 应用的性能缺陷主要是由于应用程序设计、数据库访问策略或系统资源配置等方面存在的不足所导致的。如内存泄漏、Tomcat 连接数配置问题、JVM 配置问题等。

6. Web 应用性能缺陷模式

Web 应用性能缺陷模式是对 Web 应用性能缺陷的综合观察和判断，一般包括 Web 应用性能缺陷的特性、多个性能指标以及性能指标所要满足的条件或约束。性能缺陷模式用于对 Web 应用性能方面的缺陷进行分类，主要使用可感知的性能指标和相应的判断准则。一般而言，性能缺陷模式所定义的性能缺陷具有一定的普遍性和规律性，因此可以进行自动的性能诊断。

对于 Web 应用来说，由于其多层次的体系结构以及用户访问行为的多变性，性能缺陷

的表现具有多样性特点，往往需要从多个角度来综合表示和分析 Web 应用性能缺陷模式。

7. Web 应用性能诊断

Web 应用性能缺陷诊断是在给定一组性能缺陷模式和一组日志数据后，通过构建相应的分析算法，按照性能缺陷模式所确定的性能指标和相应的条件对日志数据进行分析，从而判断哪些性能缺陷模式适用于给定的日志所反映的性能缺陷的过程。有时，往往有多种性能缺陷模式都适用于应用的性能问题。

对于每种 Web 应用性能缺陷模式，需要设计相应的实验来收集日志数据，以便验证性能诊断方法的效果。而对于给定的多种 Web 应用性能缺陷问题模式来说，它们之间往往存在相同的观察变量，导致性能缺陷诊断的难度加大。

1.3.2 研究思路

Web 应用在运行过程中会产生海量的时间序列数据，包括日志数据和执行轨迹数据。就日志数据而言，通常包括运行容器或平台（如 Tomcat、JBoss 等）所产生的日志，以及开发人员为了分析软件运行状况通过程序插装而产生的日志（简称为运行事件日志）。前者往往具有标准化的结构（如 W3C 定义的日志格式标准），而后者则在结构上具有相当大的随意性。

目前 Web 应用主要运行于虚拟机平台上，如 JVM（Java Virtual Machine）和. NET Framework，因此可以通过虚拟机提供的执行剖析（profiling）接口在软件运行时同步获得执行轨迹数据。不难看出，解决 Web 应用性能度量和分析的关键是如何分析其运行期间产生的海量时间序列数据，发现影响软件性能的指标要素以及用户访问行为，并依据这些影响因素，发现并建立它们与 Web 应用性能缺陷之间的关系，当发生性能缺陷的时候，通过对这些要素的分析，来最终定位软件的性能瓶颈所在。

导致 Web 应用性能的影响因素很多，而且相互之间是有联系的，并不是针对某一方面的性能策略的调整就一定能够解决整个应用的性能缺陷，甚至可能导致更严重的性能缺陷。本书基于 Web 应用运行期间生成的海量的多层次的日志数据，应用数据挖掘的方法，分析性能缺陷表征模式的表示方法，并依据性能缺陷表征模式，设计和实现诊断 Web 应用性能缺陷的方法，找到性能缺陷的根源所在，从而诊断软件的性能瓶颈，改善软件的性能，图 1-4 给出了本书的性能缺陷分析和诊断的总体研究思路。

本书采取理论与实践相结合的原则进行研究，设计了大量的实验用于验证通过统计方法得到的模式、模型和算法的准确性。图 1-5 给出了本书的研究方法和技术路线示意图。

具体说来，本书采用如下的方法和技术路线来开展研究。

① 基于海量日志数据的性能度量模型研究：分析 Web 应用的特点，提取性能特征并定义性能度量方法；分析 Web 应用的运行环境，进行多层次日志数据的分析和获取方法研究；面向 Web 应用的诊断目标，综合分析性能度量指标，建立 Web 应用性能度量模型。

② Web 应用性能缺陷模式：研究 Web 应用性能的度量和评价标准；进行 Web 应用