

# 针对 B/S 架构软件性能测试的应用

虞 鹏<sup>1</sup>, 贾 婧<sup>2</sup>

摘要: 介绍在 B/S 架构下利用自动化测试工具对 Web 软件进行性能测试的具体操作方法及注意事项。

关键词: B/S; 软件测试; 性能测试

中图分类号: TP311.56 文献标识码: B 文章编号: 1673-3355 (2011) 03-0019-03

## Application of B/S Architecture Software Performance Test Yu Peng, Jia Jing

**Abstract:** The specific operating procedure and precautions of the performance test about Web software by using automatic test tool under the B/S architecture software are introduced in the paper.

**Key words:** B/S; software test; performance test

企业软件建设模式为自主开发与合作开发相结合, 同时还需外购部分商业软件。这些软件的整体性能能否满足企业需求, 能否承受企业最大并发用户数等等问题一直困扰着企业信息管理部门。而随着企业信息化发展, 基于 B/S 架构的软件正在成为现有软件模式的主流。

本文就是基于这种需求, 为了加强软件项目管理, 理顺软件项目管理流程, 提高软件质量水平, 从日常的软件性能测试工作中总结出一套针对 B/S 架构主流软件的性能测试方法。

## 1 性能测试概述

### 1.1 概念

性能测试是利用成熟先进的自动化软件测试技术在正常环境与极端环境下模拟多种负载对软件系统进行各项性能指标的监测与分析。通常性能测试包括负载测试与压力测试。负载测试是在正常环境下将负载逐渐增加, 记录分析系统各项性能指标的变化情况; 压力测试是在系统、网络资源匮乏的极端条件下对内存、CPU、硬盘空间和网络带宽占有率等进行集中监控。

### 1.2 性能测试的目标

(1) 基本目标 首先测出应用系统在常规环境下总体性能参数, 主要包括每秒 HTTP 响应数、用户负载影响、每秒用户向服务器请求相应的往返延迟、吞吐量等;

(2) 高级目标 找出应用系统可能存在的性能瓶颈或者缺陷, 确认其是否可以达到用户的使用需求。整理测试结果并分析产生缺陷原因, 形成测试报告, 为软件开发及改进提供依据。

## 2 受测项目概述

“办公自动化系统”是利用信息技术改变过去复杂、低效的手工办公方式, 提高办公效率, 进而实现办公电子化、自动化的软件系统 (见图 1)。该系统基于工作流的概念, 能使企业内部人员方便快捷地共享信息, 高效地协同工作。该系统由个人办公、网上办公、公共事务、信息交流、办公指南、系统设置等 10 个模块组成, 是利用 Java 语言开发的典型三层 B/S 架构 MIS 系统, 其中间层是业务逻辑层, 应用服务器运用 WebLogic 中间件处理所有的业务逻辑, 底层数据库应用 ORACLE 10G, 中间件和数据库分别部署在两台 IBM 服务器上

1. 中国第一重型机械股份公司信息技术部助理工程师, 黑龙江 富拉尔基 161042
2. 中国第一重型机械股份公司营销事业部助理工程师, 黑龙江 富拉尔基 161042



(见表 1)。

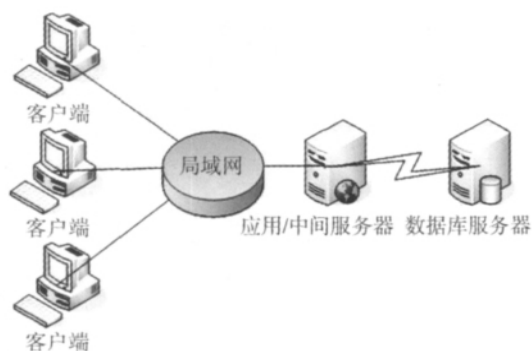


图 1 “办公自动化系统”系统架构图

表 1 服务器参数

型号	主机 IP	CPU	CPU 数
IBMX3650 M2	172.16.1.222	至强 5520	8
IBM P570	172.16.1.18	1.65	16
内存	操作系统	用途	
32G	Linux	中间件服务器	
64G	AIX	数据库服务器	

### 3 测试流程概述

(1) 先用测试工具软件录制测试脚本，将每个人机交互动作如“提交”、“查询”、“保存”等对应为测试工具的每个事务。在脚本中必须准备足够的测试数据；

(2) 利用测试工具模拟 100 个不同用户登录系统。并发进行登录系统的操作，操作相应的功能模块跟踪事务的响应时间，每秒的点击率以及整个场景中通过的事务，最终对测试结果进行分析（见图 2）。

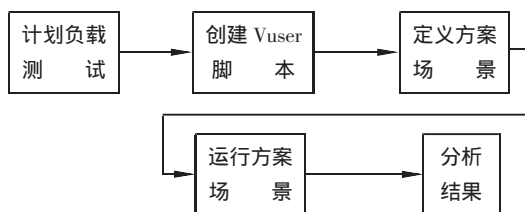


图 2 测试流程图

## 4 测试步骤及要点

### 4.1 创建 Vuser 脚本

(1) 选择适当的协议。由于测试目标设备是 web 服务器，所以此次选择 Https 协议；

(2) 录制方式一般选择 HTML-Based Script，

应用程序中包含 JavaScript 脚本且产生请求，基于浏览器的应用程序使用 Https 协议；

(3) 默认设置记录的浏览器为 IE；

(4) 如果想测定某个操作的响应时间，可以在脚本中插入事务，使用事务把该操作包装起来。分析执行结果的时候可以查看到该事务的响应时间；

(5) 插入集合点，可以使多个用户并发进行同一操作，提高操作的并发程度，以对服务器增加负载，测试并发能力；

(6) 在 Run-Time Setting 设置中，设置网络带宽以模拟不同带宽的网络；

(7) 对脚本进行参数化，设置参数变更方式；

(8) 关联脚本。

### 4.2 定义方案场景

(1) 设置场景主要包括选择脚本、设定执行用户数、选择测试负载机、设置脚本执行的方式、设置集合点、设置 Run-Time Setting；

(2) 如果要模拟的用户数比较多，则应该设置多台测试负载机，一般主流的 PC 机至少应能模拟 100 个用户对服务器的访问；

(3) 设置脚本的执行方式包括设置用户的启动方式和终止方式；

(4) 设置集合点包括开关集合点及设置用户通过集合点的方式；

(5) 设置 IP 欺骗。如果服务器对用户的 IP 有限制（安全原因），启用 IP 欺骗；

(6) 设置服务器监控计数器应包含：内存，CPU，线程，进程，网络，磁盘。

### 4.3 分析结果

(1) 查看分析结果中事务响应时间、服务器平均吞吐量，执行用户人数等；

(2) 查看线程图，用户变化图，响应时间图，吞吐量图；

(3) 查看服务器监控的计数器图；

(4) 分析各个性能指标是否符合需求；

(5) 分析各个曲线图是否存在异常情况；

(6) 有时候分析单个要素并不能反映存在的问题，需要多个要素联系起来进行分析。

## 5 测试结论

通过模拟 100 个不同用户登录系统，每隔 5 s 增加一个用户，达到峰值并运行一段时间后每 5

s 减少一个用户直至结束,运行总时间 27 min, 每秒钟平均点击次数为 276 次, 每秒 HTTP 响应数平均值 16 个, 平均吞吐量为 8601765.306 字节; 系统在压力测试过程中主要是 IST 系统 CPU 利用率较小, 当并发达到 100 时, 系统 CPU 利用率就达到 30%, 数据库 I/O 指标正常。

### 5.1 负载用户影响

负载用户影响图用于显示在负载测试过程中运行的虚拟用户数与平均交易响应时间之间的关系(见图 3)。可以看出在虚拟用户负载从 0~100 的渐进过程中平均交易响应时间在 12~22 s 变动, 从第 5 个用户到第 20 个用户的平均交易响应时间基本稳定在 12 s 左右, 增加到第 89 个时交易时间最长。事务响应时间基本成一条曲线, 说明系统性能基本稳定。

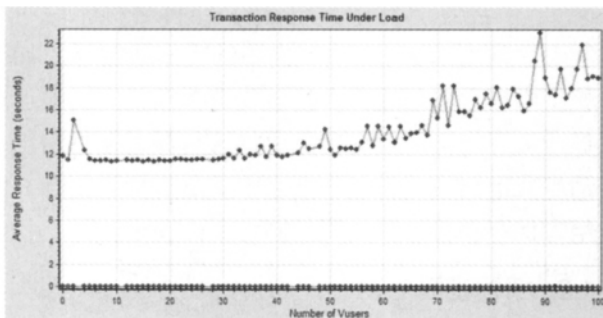


图 3 负载用户影响图

### 5.2 每秒钟的点击次数图

“每秒点击次数”图用于显示在负载测试场景运行期间的每一秒内 Vuser 向 Web 服务器发出的 HTTP 请求数(见图 4)。在该项目测试中平均每秒钟点击次数 276 次, 在第 12 s 时达到 340 次/s 峰值。

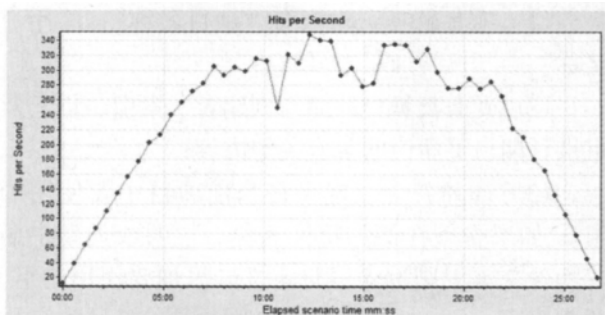


图 4 每秒钟的点击次数图

### 5.3 每秒 HTTP 响应数图

“每秒 HTTP 响应数”图用于显示负载测试场景运行期间每秒从 Web 服务器返回的 HTTP 状态代码数(按状态代码分组)(见图 5)。而 HTTP

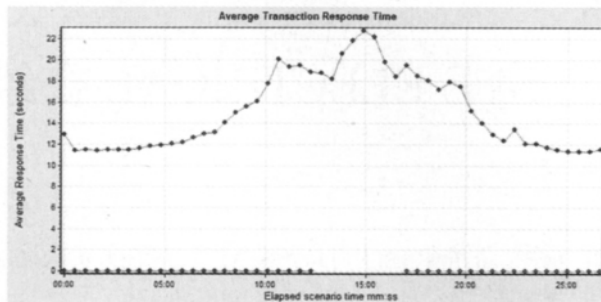


图 5 每秒 HTTP 响应数图

状态代码指示 HTTP 请求的状态。例如, “请求成功”、“页面未找到”等等。图中 X 轴表示自场景开始运行以来经过的时间, Y 轴表示每秒 HTTP 响应数。在场景开始运行后的第 15 s 达到最大响应数 27, 而平均值为 16。

### 5.4 吞吐量

“吞吐量”图显示在负载测试场景运行的每一时刻, 服务器上的吞吐量。吞吐量以字节为单位, 表示 Vuser 在任意给定的一秒内从服务器接收的数据量。本项目测试根据服务器吞吐量对 Vuser 生成的负载量进行评估(见图 6)。以 X 轴表示自场景开始运行以来经过的时间, Y 轴表示服务器的吞吐量(字节)。测试结果表明在场景开始运行后的第 55s 出现最大吞吐量, 11 000 000 字节, 而平均吞吐量为 8 601 765.306。

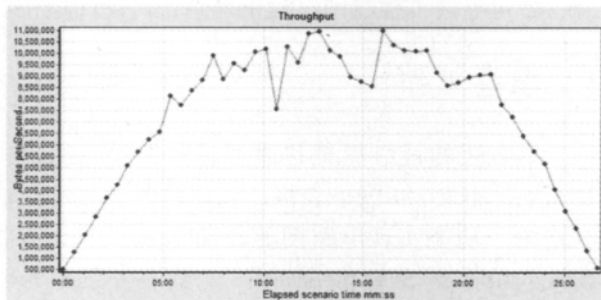


图 6 吞吐量图

## 6 结 语

本文利用基于 B/S 架构软件的性能测试方法对“办公自动化系统”进行的实际测试, 既为评价该软件的质量及进行软件验收提供的准确依据; 又在具体实施过程中对测试方法的使用规律及要点有了更深刻的认识和更全面的掌握。如今, 该性能测试方法已经在企业的软件建设中普及推广, 为使今后企业信息化工作更加深入, 对 B/S 架构软件性能测试方法还需要做进一步深入的研究。

收稿日期: 2011-04-22