

软件性能测试需求的获取方法综述

温艳冬

(大连东软信息学院, 辽宁 大连 116023)

摘 要: 性能测试需求的质量直接影响性能测试的效果, 在分析Web应用系统性能测试目的的基础上, 提出性能测试需求描述要达到准确、一致和特定的要求, 进一步明确性能测试需求必须要确定4W1H, 即性能测试的需求必须包含where, what, when, who和how, 并综述了几种有效的获取性能测试需求的方法。

关键词: 性能测试; 测试需求; 需求获取

1 引言

基于Web服务器的应用系统由于提供浏览器界面而无须安装, 大大降低了系统部署和升级成本而得以普遍应用。目前, 很多企业的核心业务系统均是Web应用, 但当Web应用的数据量和访问用户量日益增加, 系统不得不面临性能和可靠性方面的挑战。因此, 无论是Web应用系统的开发商或最终用户, 都要求在线前对系统进行性能, 科学评价系统的性能, 从而降低系统上线后的性能风险。

在很多性能测试项目中, 由于不能合理定义系统的性能测试需求, 不能建立和真实环境相符的负载模型, 不能科学分析性能测试结果, 导致性能测试项目持续时间很长或不能真正评价系统性能并提出性能改进措施。因此, 性能测试需求分析的正确性是整个性能测试工作的最基本前提。若不能保证性能测试需求分析的正确性, 即使性能测试工具使用的再正确, 性能测试执行的再顺利, 也无法保证性能测试达到预期的效果, 即可能无法发现性能瓶颈、或者发现不了实际情况中应该出现的瓶颈。

本文从分析性能测试的目的出发, 提出性能测试需求描述的三个要求, 综述性能测试需求的获取常用方法。

2 性能测试的目的

性能测试的目的不仅是发现软件缺陷, 还可能包

括以下几个方面:

(1) 验证能力。这是性能测试中最简单也是最常用的一个应用领域, 典型的能力验证问题会采用这样的描述方式: “***系统能够在***条件下具有***能力? ”。通常情况下, 企业在进行项目验收阶段要求能力验证型的性能测试或者委托第三方软件测试机构开展独立的性能验证, 其主要特点是在已确定的生产环境中实际使用被测系统, 即这套系统能不能承受大量的并发用户同时访问? 常以典型场景设计测试方案和用例。

(2) 规划能力。这与(1)有较大的不同, 以规划能力为目的的性能测试关注的是“应该如何才能使系统具有要求的性能能力?”或者“系统能否支持未来一段时间内的用户增长?”, 因此, 这种性能测试强调对系统当前性能的评估, 通过评估可以在应用实际部署之前, 预见系统负载压力的承受能力。

(3) 调优性能。性能调优是以第一种或第二种为目的的性能测试实施后提供原始数据进而分析系统瓶颈和优化为目的, 因此(3)常与其他的性能测试活动交杂在一起。该类性能测试需要在确定的基准环境下, 采用基准负载, 关注基准性能指标后, 调整系统运行环境和实现方法, 执行测试, 记录测试结果进行分析, 再调整、执行、分析, 不断往复, 直到系统性能达到要求为止。比如: 用户提出业务操作响应时间长, 如何定位问题, 调整性能?

3 性能测试需求描述要求

(1) 准确。如**系统必须在不超过 10 秒的响应时间内, 处理 20 起登录和注销系统任务。再如**搜索时间最大不超过5秒以及平均时间在1~3秒以内。

(2) 一致。开发工程师、用户和性能测试工程师对有关术语的理解要一致, 如: 并发用户数、动态用户数、静态用户数:

静态用户(注册用户): 3500以上

动态用户(在线用户): 1500以上

并发用户: 500以上

(3) 特定。性能测试的需求一定是有条件的, 如:

检查系统在 200 个用户的负载下, 所有业务动作是否可用及稳定;

检查系统在300 个用户的负载下, 连续运行 72 小时过程中, 订单上传、转单、详情单查询、发运等业务动作是否可用及稳定;

检查系统在 8.0 GB 业务数据、1500 个用户、500 个并发用户运行的负载下, 连续运行 72 小时过程中, 以上业务动作是否可用及稳定。

因此, 性能测试需求必须要包含有多少用户(who)在什么时间(when)或者持续多久(when)进行了什么业务(what), 最终需要关注怎样的指标(how)。除此以外, 需要根据项目性质和性能测试的目标来获得性能测试需求的来源(where), 归纳为4W1H。下面将介绍一些常用的性能测试需求获取的方法。

4 获取性能测试需求的方法

性能测试需求是应用需求的衍生, 既需要借助于相关的理论知识, 又要依靠测试工程师在相关领域的经验积累, 根据前面4W1H的性能测试需求的要求, 即对性能测试需求进行整理, 确定恰当的并发用户数、在适当的时间进行典型的业务活动时, 关注的性能指标有怎样的结果, 笔者根据多次性能测试实践经验, 总结获取性能测试需求的方法如下。

4.1 性能需求信息来源 (where)

(1) 开发过程相关文档。这是性能测试需求的主要来源, 项目开发计划书、需求规格说明书、设计说明书、测试计划等文档都可能涉及性能测试的要求, 通过收集这些资料, 可以找到初步的性能需求。相关的项目干系人有客户代表、项目经理、需求分析员、系统架构设计师、产品经理等。

(2) 相似项目性能需求。公司的其他产品或项目会累积出一些数据, 如: **技术论坛一小时最多能发1000新帖; ****博客平均每天新增800篇, 以这些数据为确认新项目测试需求的基础。

(3) 业界公认标准。如响应时间, 根据服务器的不同和项目的具体情况可能有两类标准:

A类标准——

4秒以内, 用户可以接受

4-9秒, 30%用户离开

8-10秒, 60%用户离开

超过10秒, 90%用户离开

B类标准——

8秒, 用户可接受

16秒, 50%用户离开

32秒, 90%用户离开

(4) 用户使用模型。性能测试要通过一系列场景的执行来完成, 分析用户的使用模型是获取性能测试需求的有效手段, 即定义系统的典型使用方式, 考虑哪些用户使用系统的哪些典型业务, 在什么时间段和用户数量的估计值, 因此需要和最终的用户有很好的沟通, 最好能够实地考察用户的应用情况。如某OA系统的每天早上8:00会有200个用户在10分钟内登录系统; 每天查询交易的高峰是在9:00-11:00和下午的14:00-16:00等。

4.2 80~20原则估算测试强度 (how)

80~20原理: 每个工作日中80%的业务在20%的时间内完成。

举例:

每年业务量集中在8个月, 每个月20个工作日, 每

个工作日8小时即每天80%的业务在1.6小时完成。去年全年处理业务约100万笔，其中15%的业务处理中每笔业务需对应用服务器提交7次请求；其中70%的业务处理中每笔业务需对应用服务器提交5次请求；其余15%的业务处理中每笔业务需对应用服务器提交3次请求。根据以往统计结果，每年的业务增量为15%，考虑到今后3年业务发展的需要，测试需按现有业务量的两倍进行。

每年总的请求数为：

$$(100 \times 15\% \times 7 + 100 \times 70\% \times 5 + 100 \times 15\% \times 3) \times 2 = 1000$$

万次/年

每天请求数为：

$$1000 / (20 \times 8) = 6.25 \text{ 万次/天}$$

每秒请求数为：

$$(62500 \times 80\%) / (8 \times 20\% \times 3600) = 8.68 \text{ 次/秒}$$

即服务器处理请求的能力应达到9次/秒。

4.3 任务分布图 (what + when + who)

任务分布图是以一种直观的方式展示性能测试需求，描述一些交易任务在24小时内的交易情况，重点关注并发用户的数量以及典型的业务操作。如表1是某网上书店的任务分布图，从中可见：12：00~16：00的交易混合程度较高，“预定图书”任务的并发用户在14：00达到最大。

如果需要进一步了解典型任务的平均值、峰值以及对web服务器、数据库服务器的压力等信息，可以建立交易混合表。

表1 某网上书店的任务分布图

典型任务	并发用户数											
登 录				30	50	100	110	60	40			
查 询					20	80	70	30	20			
预 定						30	500	20	10			
时 间	2	4	6	8	10	12	14	16	18	20	22	24

4.4 web服务器访问日志

一般的web server有两部分日志：一是运行中的日志，它主要记录运行的一些信息，尤其是一些异常

错误日志信息；二是访问日志信息，它记录访问的时间、IP、访问的资料等相关信息。为了获取系统性能测试的需求，可以分析web server的访问日志以了解系统更多真实负载和主要的业务场景。下面介绍一下利用tomcat产生的访问日志数据所做的有效分析。

首先是配置tomcat访问日志数据，默认情况下访问日志没有打开，配置的方式如下：

编辑 \${catalina}/conf/server.xml文件：

(注:\${catalina}是tomcat的安装目录)

把以下的注释(<!-- -->)去掉即可。

<!--

<Valve className="org.apache.catalina.valves.AccessLogValve"

directory="logs"

prefix="localhost_access_log." suffix=".txt"

pattern="common" resolveHosts="false"/>

-->

其中 directory是产生的目录 tomcat安装\${catalina}作为当前目录；pattern表示日志生产的格式，common是tomcat提供的一个标准设置格式。其具体的表达式为 %h %l %u %t "%r" %s %b。通过这个配置能得到的数据有：

%h 访问的用户IP地址

%l 访问逻辑用户名，通常返回'-'

%u 访问验证用户名，通常返回'-'

%t 访问日时

%r 访问的方式(post或者是get)，访问的资源和使用

的http协议版本

%s 访问返回的http状态

%b 访问资源返回的流量

%T 访问所使用的时间

有了这些数据，可以根据时间段做以下的分析处理：

(1) 独立IP数统计

(2) 访问请求数统计

(3) 访问资料文件数统计

- (4) 访问流量统计
- (5) 访问处理响应时间统计
- (6) 统计所有404错误页面
- (7) 统计所有500错误的页面
- (8) 统计访问最频繁页面
- (9) 统计访问处理时间最久页面
- (10) 统计并发访问频率最高的页面

如图1所示, 将系统运行一段时间后获取的数据分析汇总后形成的图示, 为性能测试工程师提供了非常有价值的信息, 从图中可见, 并发用户数在7:00-11:00之间明显增大, 平均值在40左右。

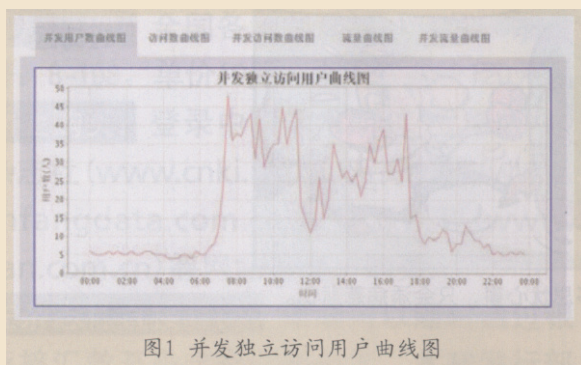


图1 并发独立访问用户曲线图

4.5 UCMLTM

UCMLTM (User Community Modeling Language) 是一个符号集合, 这些符号可以创建虚拟系统用法模型, 以及描述相关参数。当把它应用到负载压力性能测试时, 这些符号可用于表示工作量分配、操作流程、重点工作表、矩阵和马尔可夫链等。负载压力性能测试工程师在决定测试中用到什么活动, 以及它们发生的频率时, 经常用到这些参量。

通常应用SmartDraw 或者 Microsoft visio 绘制 UCML, 进行负载压力测试需求分析。UCML的数据来源有两种方式: 一是通过与最终用户的沟通, 详细询问应用情景, 根据一定的常识推理得到; 二是通过分析已有的数据, 如数据库的日志, web server 的访问日志等获得。UCML的好处在于提供了一种易于理解、便于沟通的表现形式, 尤其在应用自动化性能测试工具时, 方便性能测试计划、分析、设计和实施人

员的沟通。图2是一个在线书店的 UCMLTM 图表, 为负载压力测试提供了需求。

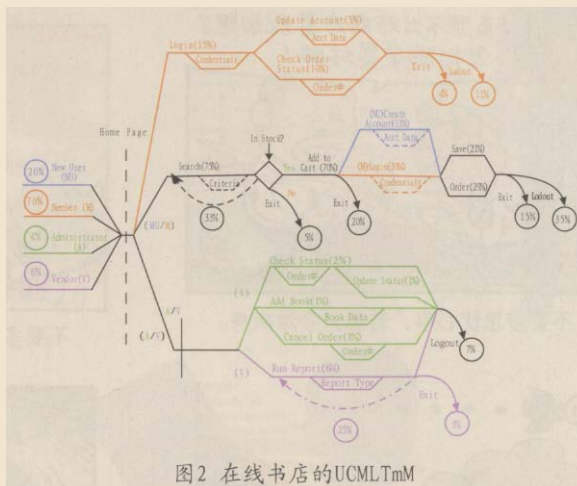


图2 在线书店的UCMLTmM

5 总结

Web应用项目的性能测试成功的关键不在于性能测试工具, 而在于有效的性能测试分析方法和实践。只有切实掌握性能测试需求分析方法, 性能测试实践经验, 才能保证一个Web应用性能测试的成功。基于4W1H的性能测试需求描述标准能够为获取有效的性能测试需求提供一个依据, 结合性能测试目的而选取适用的性能测试需求获取方法才是有效的。■

参考文献

- [1] 柳纯录, 黄子河, 陈绿萍. 软件评测师教程[M]. 北京: 清华大学出版社, 2005.
- [2] 赵艳玲. 软件开发中的需求获取[J]. 现代电子技术, 2007 (11): 580-581.
- [3] 冯铮. 基于WEB的软件测试规范与质量标准体系的研究[J]. 计算机与数字工程, 2007 (12): 71-74.
- [4] 谈谈辰, 尹军. 基于负载的WEB性能测试研究与应用[J]. 现代电子技术, 2007 (14): 139-142.
- [5] 常广炎. 基于WEB系统的性能测试[J]. 办公自动化, 2008 (2): 25-27.

- 欢迎投稿、发表任何读后感、意见或建议。
- 编辑: 王飞 wangfei@neusoft.edu.cn