



工业和信息化
人才培养规划教材

Industry And Information
Technology Training
Planning Materials

软件测试项目实战 之性能测试篇

Software Test Project

杨焰 简显锐 胥林 © 主编

张渝 秦文生 赵福贵 © 副主编

- + 采用**项目教学法**，学生在实际项目中学习软件
测试知识
- + **校企合作**编写教材，体现企业**软件测试岗位**的
技能要求



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



工业和信息化
人才培养规划教材

Industry And Information
Technology Training
Planning Materials

软件测试项目实战 之性能测试篇

Software Test Project

杨焰 简显锐 胥林 © 主编

张渝 秦文生 赵福贵 © 副主编

人民邮电出版社
北京

图书在版编目 (C I P) 数据

软件测试项目实战之性能测试篇 / 杨焰, 简显锐, 胥林主编. — 北京: 人民邮电出版社, 2016.9
工业和信息化人才培养规划教材
ISBN 978-7-115-39829-1

I. ①软… II. ①杨… ②简… ③胥… III. ①软件—测试—高等学校—教材 IV. ①TP311.5

中国版本图书馆CIP数据核字(2015)第152156号

内 容 提 要

本书采用项目教学方法, 围绕一个真实的腾图办公自动化(OA)系统平台项目展开测试任务。主要内容包括 OA 系统项目性能测试分析, 项目性能环境与数据准备, 项目性能测试脚本开发, 项目负载分析、调度、运行调度, 项目测试结果分析, 性能测试报告等; 重点是运用 IBM 公司的 RPT 自动化性能测试工具完成了 OA 项目相关功能模块的自动化测试内容。

本书既可作为高等院校软件测试专业的教材, 又可作为社会培训机构的培训教材, 同时也适合从事软件测试工作的读者自学参考。

◆ 主 编 杨 焰 简显锐 胥 林

副 主 编 张 渝 秦文生 赵福贵

责任编辑 马小霞

责任印制 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京鑫正大印刷有限公司印刷

◆ 开本: 787×1092 1/16

印张: 12

2016 年 9 月第 1 版

字数: 270 千字

2016 年 9 月北京第 1 次印刷

定价: 39.80 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

前言 FOREWORD

目前我国软件的规模日益增大，企业对软件产品的质量要求越来越高，对软件测试人员的需求也越来越大。为了满足企业对软件测试人才的需求，各级各类院校纷纷开设了软件测试的课程。我们通过多年的教学研究发现，目前的教材理论知识过多，学生的技能没有达到企业软件测试岗位要求，知识与技能未能很好地融合，因此组织院校老师与企业工程师共同打造了这本校企合作教材，旨在使软件测试教学更符合学生的认知规律，提高学生学习质量与效率，以期培养出具备良好职业素养的专门人才。

本书采用项目教学模式，以职业活动和工作过程为导向，以一个复杂、大型、综合的项目为基础，实现知识、理论和实践的整体设计，是一本具有综合性、实战性的一体化教材。教材中的项目是经过作者学校专业教师团队与企业工程师多次交流讨论后，从企业已经完成的若干个项目选择的，具有代表性，企业工程师全程参与了教材的编写过程。

本书是同系列书《软件测试项目实战之功能测试篇》的姊妹篇，重点讲性能测试的工具使用，其中测试工作任务载体 OA 系统的分析和介绍可以参看《软件测试项目实战之功能测试篇》一书。

本书内容和结构的安排基于企业真实的软件测试工作。第 1 章主要介绍性能测试的一些基础知识和概念，包括性能测试典型目标、性能测试流程、性能测试分类等，然后引入 IBM 公司性能测试工具 RPT。第 2 章介绍通过 RPT 工具创建一个性能测试的流程，在 OA 系统上通过录制方式生成性能测试脚本。第 3 章~第 7 章主要介绍 RPT 工具在 OA 系统测试过程中的主要功

能设置与选择,包括 HTTP 协议和在 RPT 工具调试脚本时人工判断脚本实际运行结果方法,RPT 工具自动化判断实际结果方法,RPT 工具数据驱动框架,RPT 工具关联数据方法,RPT 工具性能测试脚本扩展,RPT 工具创建一个调度、配置调度、运行调度。第 8 章综合前面的学习,介绍运用 RPT 工具对 OA 系统的考勤功能进行实战操作。第 9 章是介绍以 OA 系统测试环境的搭建为基础介绍性能测试环境的搭建方法。

本书由杨焰、简显锐和胥林任主编,张渝、秦文生和赵福贵任副主编。其中杨焰编写第 1 章和第 6 章,简显锐编写第 4 章和第 5 章,胥林编写第 7 章,张渝编写第 8 章,秦文生编写第 9 章,赵福贵编写第 2 章和第 3 章。本书在编写过程中参考了许多文献及成果,对本书参考书籍的作者表示深深的敬意和诚挚的感谢。由于编者水平有限,错漏之处在所难免,敬请广大读者、专家批评指正。

编者

2016年5月

第1章 性能测试基础 1

1.1 性能测试概念	2	1.8 RPT 工具概述	11
1.2 性能测试作用	3	1.8.1 基于 Eclipse 平台	11
1.3 性能测试指标	4	1.8.2 RPT 性能测试脚本	11
1.4 性能测试流程	5	1.8.3 性能测试负载模型	12
1.5 性能测试的分类	7	1.8.4 测试结果	14
1.6 性能测试工程师技能模型	8	1.9 OA 系统分析	14
1.7 性能测试工具介绍	9	1.10 总结	18

第2章 创建测试 19

2.1 测试创建简介	20	2.3 录制测试	25
2.2 创建一个性能测试项目	22	2.4 总结	31

第3章 测试验证点 32

3.1 HTTP 协议	33	3.4 添加验证点	44
3.1.1 HTTP 协议简介	33	3.4.1 设定页面标题验证点	45
3.1.2 HTTP 请求	33	3.4.2 设定响应代码验证点	47
3.1.3 HTTP 响应	37	3.4.3 设定响应大小验证点	50
3.2 人工判断	41	3.4.4 设定内容验证点	52
3.3 自动化判断	43	3.5 总结	54

第4章 数据驱动测试 55

4.1 数据驱动测试简介	56	4.3.2 为测试添加数据池	68
4.2 RPT 的数据池	57	4.3.3 将测试中的变量与数据池中 的列进行关联	72
4.3 OA 系统中使用数据驱动测试	58	4.4 总结	77
4.3.1 创建并编辑数据池	59		

第5章 数据关联 79

5.1 OA 系统数据关联	80	5.3 数据关联的调整	91
5.2 OA 系统使用关联	82	5.4 总结	95

CONTENTS

目录

第 6 章 扩展测试 96

6.1 添加事务处理	97	6.3 添加循环	100
6.2 添加条件逻辑	98	6.4 总结	101

第 7 章 负载和调度 102

7.1 用户组和调度的概念	103	7.3.5 设置思考时间行为	115
7.2 用户组	104	7.3.6 设置运行持续时间	116
7.3 为 OA 系统创建调度	105	7.4 运行调度	117
7.3.1 创建一个调度	106	7.4.1 设置一个启动配置	117
7.3.2 增加元素到一个调度中	107	7.4.2 运行一个调度	121
7.3.3 设置启动运行的用户数量	113	7.5 总结	124
7.3.4 在不同的时间启动用户	114		

第 8 章 性能测试实战 125

8.1 性能测试需求	126	8.3 性能测试开发	130
8.2 性能测试计划	126	8.3.1 创建性能测试项目	130
8.2.1 概述	126	8.3.2 录制测试	131
8.2.2 测试目标	127	8.3.3 脚本优化	137
8.2.3 测试设计	127	8.4 添加性能调度	142
8.2.4 数据统计	129	8.5 测试结果分析	145
8.2.5 性能测试报告输出	129	8.6 总结	150
8.2.6 性能调优与回归	130		

第 9 章 OA 系统测试环境搭建 151

9.1 测试环境搭建流程	152	9.2.4 JDK 安装与配置	156
9.2 测试环境搭建	152	9.2.5 Mysql 安装与配置	163
9.2.1 测试环境配置要求	153	9.2.6 Tomcat 安装与配置	173
9.2.2 硬件需求配置	155	9.2.7 被测试应用程序部署	178
9.2.3 操作系统安装	155	9.3 总结	185

CONTENTS

目录

第 6 章 扩展测试 96

6.1 添加事务处理	97	6.3 添加循环	100
6.2 添加条件逻辑	98	6.4 总结	101

第 7 章 负载和调度 102

7.1 用户组和调度的概念	103	7.3.5 设置思考时间行为	115
7.2 用户组	104	7.3.6 设置运行持续时间	116
7.3 为 OA 系统创建调度	105	7.4 运行调度	117
7.3.1 创建一个调度	106	7.4.1 设置一个启动配置	117
7.3.2 增加元素到一个调度中	107	7.4.2 运行一个调度	121
7.3.3 设置启动运行的用户数量	113	7.5 总结	124
7.3.4 在不同的时间启动用户	114		

第 8 章 性能测试实战 125

8.1 性能测试需求	126	8.3 性能测试开发	130
8.2 性能测试计划	126	8.3.1 创建性能测试项目	130
8.2.1 概述	126	8.3.2 录制测试	131
8.2.2 测试目标	127	8.3.3 脚本优化	137
8.2.3 测试设计	127	8.4 添加性能调度	142
8.2.4 数据统计	129	8.5 测试结果分析	145
8.2.5 性能测试报告输出	129	8.6 总结	150
8.2.6 性能调优与回归	130		

第 9 章 OA 系统测试环境搭建 151

9.1 测试环境搭建流程	152	9.2.4 JDK 安装与配置	156
9.2 测试环境搭建	152	9.2.5 Mysql 安装与配置	163
9.2.1 测试环境配置要求	153	9.2.6 Tomcat 安装与配置	173
9.2.2 硬件需求配置	155	9.2.7 被测试应用程序部署	178
9.2.3 操作系统安装	155	9.3 总结	185



第 1 章 性能测试基础

随着当今软件行业的蓬勃发展，软件与企业的运营、个人用户的生活联系愈加紧密，软件的应用给企业和个人都带来了很大的帮助。但同时用户对软件的要求也愈加苛刻，用户对软件厂商提供的软件产品已经不仅仅局限于正常功能的使用，对软件产品的安全性、易用性、可靠性等方面都有着一定的要求。软件产品任何一方面不能满足用户的需求，可能都会影响用户的体验，都可能面临着用户的抛弃。本书将要描述的性能测试就是用户关注的重要指标之一，是软件质量衡量的重要的标准之一。

- 学习目标：能够深入理解和掌握本章的性能测试理论。
- 知识点：性能测试作用；性能测试的指标；性能测试流程；性能测试分类。

1.1 性能测试概念

性能测试是指使用一定的手段判定软件系统在一个给定的环境和场景中的性能表现是否与预期目标一致，评判系统是否存在性能缺陷，并根据测试结果识别性能瓶颈，改善系统性能的过程。

性能测试主要用于验证软件系统的实际性能指标是否满足用户的期望指标。性能测试概念中的两个前提条件非常重要，那就是一定的环境和场景。因为软件的性能指标和它运行的环境和场景是紧密相关的。

1. 运行环境

系统的性能与它所处的运行环境关系很大。例如，常说法拉利跑车性能出众，但这个说法实际上默认了适合它的运行环境，如高速公路或者专业的赛道。如果把法拉利跑车放到一个乡村的山路上运行，它可能跑不过一辆拖拉机，因为法拉利底盘低，可能陷到坑中无法运行。同样的道理，同一个软件系统放到不同的环境下，表现出的性能也可能有天壤之别。

影响性能测试环境的环境因素是多方面的，如使用的浏览器、网络带宽、操作系统、Web 服务器、应用服务器、硬件服务器、数据库等。

2. 使用场景

性能测试的另外一个前提条件就是一定的场景，也就是说一个系统的性能表现与用户如何使用也是有很大关系的。例如，春运时去 12306 网站购买火车票时，能够感受到 12306 系统的性能满足不了用户的需求，但是如在非春运时间去用 12306 系统时，这个系统使用起来还是比较不错的。系统还是一个系统，运行的环境也没有变化，体现出的性能却完全不同，原因就是用户使用的场景不同，使用方式不一样。春运的时候用户是如何去使用这个系统的呢？第一用户量非常大，第二就是用户非常集中。春运的票非常紧张，比如早晨七点放票，相信不会有用户七点半去买票，肯定都是尽量在第一时间去买票，这就导致用户非常的集中，并发性很强。而非春运时间买票，用户使用的方式就完全不同了。用户没有那么多，用户也不会集中在某一个时间区间去买票，并发性低，所以系统的压力也没有那么大，这样系统的性能就要理想很多，性能的表现完全不同。

基于以上分析，平时在谈论一个系统性能时，一定要强调系统的运行环境和运行场景，否则给出的性能指标都是不准确和不严谨的。

1.2 性能测试作用

测试工程师在对产品做性能测试时，必须要明确自己的测试目标是什么，需要验证系统的哪些方面。

1. 系统的最大容量

系统的最大容量是一个软件系统即将发布前或者一个营销活动开展前，系统在一定环境和场景下，最大能够支撑多少并发用户数，这个指标非常关键。一般产品在上架前或者营销活动开展前，相关的部门都会预估一下未来产品的最大用户数或参与活动的最大并发用户量，通过与目前的产品所能支持的最大用户数来对比，来决定产品是否上线使用或是否开展营销活动。软件产品所能支撑的最大用户数是给予产品研发人员信心的指标，就像平时去买一辆汽车一样，基本每个用户都会关心车的最高时速。这个指标实际很少用到或者根本用不到，但是用户还是关心这个指标，就是为了确认需要跑得尽可能快的时候能跑多快。

2. 常见负载下的指标

厂商关心自己研发的产品的最大系统用户量，但更加关心的是平时使用时最常见的负载下的指标是否满足要求。因为最大的并发用户量在一年的运行中极少出现或者根本就不会出现。例如，淘宝的用户高峰应该是在一年一度的“光棍节”的营销，这个用户高峰一年可能也就出现一次，但是为了这次营销活动淘宝系统必须要能支撑这个最大并发用户量。在淘宝其余的运行时间内常见的负载压力可能是最高峰的 20% ~ 40% 的用户量，研发人员应该更加关心的是这个负载压力下系统的各种性能指标，因为在这个压力状况下，用户接触最多，当然要保证这些常见压力下的性能表现。就像用户买车一样，用户平时开车最常见的时速是 60km/h、80 km/h、100 km/h，那么用户最关心的应该是这些时速下车的稳定性、油耗、噪声等方面的参数。

3. 系统的稳定性

系统经过严格的功能测试以后，基本保证了系统功能的正确性。但是有些缺陷是功能测试无法发现的，因为功能测试存在一定的局限性，例如，测试的时间比较短、测试的数据量比较小，这些都导致有些缺陷无法通过功能测试暴露出来。有些缺陷只有在长时间的运行下或在一定的大数据量下缺陷才能暴露出来，例如，内存泄露这类缺陷短时间内就很难发现。下面的代码是一段 C 语言代码。

```
#include <stdio.h>
#include <string.h>
int main(void)
```

```
{  
    char *p = (char *)malloc(100);  
    strcpy(p, "hello");  
    return 0;  
}
```

上面的函数每次执行时会在堆内存上申请 100 字节的空间，然后向这个内存上存放一段字符串“hello”。这个函数的问题在于申请了内存，没有调用 `free(p)` 进行内存的释放，也就意味着调用一次此函数，内存就会泄露 100 字节。

当用户使用某个功能时会调用上面的函数一次，调用一次泄露 100 字节，那么调用 1000 次泄露内存： 1000×100 字节=100000 字节，大约 100KB。也就意味着测试人员做功能测试时执行了此功能 1000 次也才有 100KB 字节的泄露，100KB 字节是无法通过内存曲线的观测发现的，这类曲线只能依赖性能测试才能发现。使用性能测试工具保证一定的负载压力（如最佳用户量）长时间运行（如运行 7x24h）就可以发现这类缺陷，因为在一定的负载压力下长时间运行就可以看到内存曲线不断向上的趋势。

用户在拿到一个软件时，经常看到两个指标，即最低配置和最佳配置。最低配置是指软件能够正常运行允许的最低配置，如果低于此配置就不能正常使用软件。最佳配置是指性价比最高的一种配置，既满足了系统的各种性能指标，又最大程度地降低成本。软件厂商提供的最低配置和最佳配置的依据是测试。性能测试是指在不同的软件、硬件以及网络环境配置下，通过运行一种或多种业务在一定的并发或在线用户数量情况下，获得不同配置的性能指标，用于选择最佳的设备及参数配置。

1.3 性能测试指标

1. 响应时间

响应时间是指从用户发出请求后开始计时，直到用户收到请求结果返回之间所消耗的时间。这个指标直接影响到用户体验，对于用户而言是最看中的指标之一。例如单击登录，从单击登录到登录完成返回登录成功页面需要消耗 1s，那么就说这个操作的响应时间是 1s。

2. 吞吐量

反映单位时间内能够处理的事务数目。例如，对于一个系统来说一个用户登录需要 1s，如果系统同时支持 10 个用户登录，且响应时间是 1s，那么吞吐量就是 10 个/s。这个指标反映了被测系统对请求的处理效率。

3. 服务器资源占用

服务器资源占用是指系统在性能测试过程中，被测服务器各种资源占用的情况，如 CPU 利用率、内存使用率、磁盘读写等相关资源使用指标。软件系统在保证性能指标的前提下，资源的利用率越低，说明系统越优秀。即使一个系统响应时间满足了需求，但是如果后台服务器的资源利用率过高，那这个性能测试也是一个失败的测试。因为资源利用率过高就代表着系统处于一种临近极限的状态，负载稍微增加一些就会失败，或者已经处于一种不稳定的状态。

4. 业务成功率

在做性能测试时除了响应时间、资源利用率要得到保证以外，还需要关心业务成功率。业务成功率是指在性能测试过程中业务成功和失败的比率。例如，在对系统登录功能的性能测试过程中，模拟 100 个用户，每个用户登录 100 次，共登录了 1 万次。测试完成以后测试工程师必须要统计出 1 万次的登录中成功的比率是否达到了要求，如果测试过程中登录出现了大量的失败，那这个测试也是一个失败的性能测试。

1.4 性能测试流程

在执行功能测试时，测试工程师是按照一定的流程完成项目测试的。性能测试在实施过程中也要遵守一定的流程，性能测试流程如图 1-1 所示。

1. 应用系统分析

测试一个系统的前提就是要熟悉这个系统。如果不熟悉系统，把系统测试完整是不太可能的。所以在做系统功能测试前要做功能测试需求分析，同样做性能测试前也要先做性能测试需求分析。要分析的内容如下。

- (1) 测试对象和范围。
- (2) 测试指标。
- (3) 被测系统架构、平台。
- (4) 性能测试项操作流程。

只有把以上内容明确了，才能更好地设计性能测试计划、性能测试方案、测试工具的选型等后续工作。

2. 性能测试计划

性能测试计划主要明确性能测试范围、性能测试环境、性能测试所需资源规划及筹备计划、性能测试工作项目及进度安排、性能测试出口标准、性能测试风险管理。

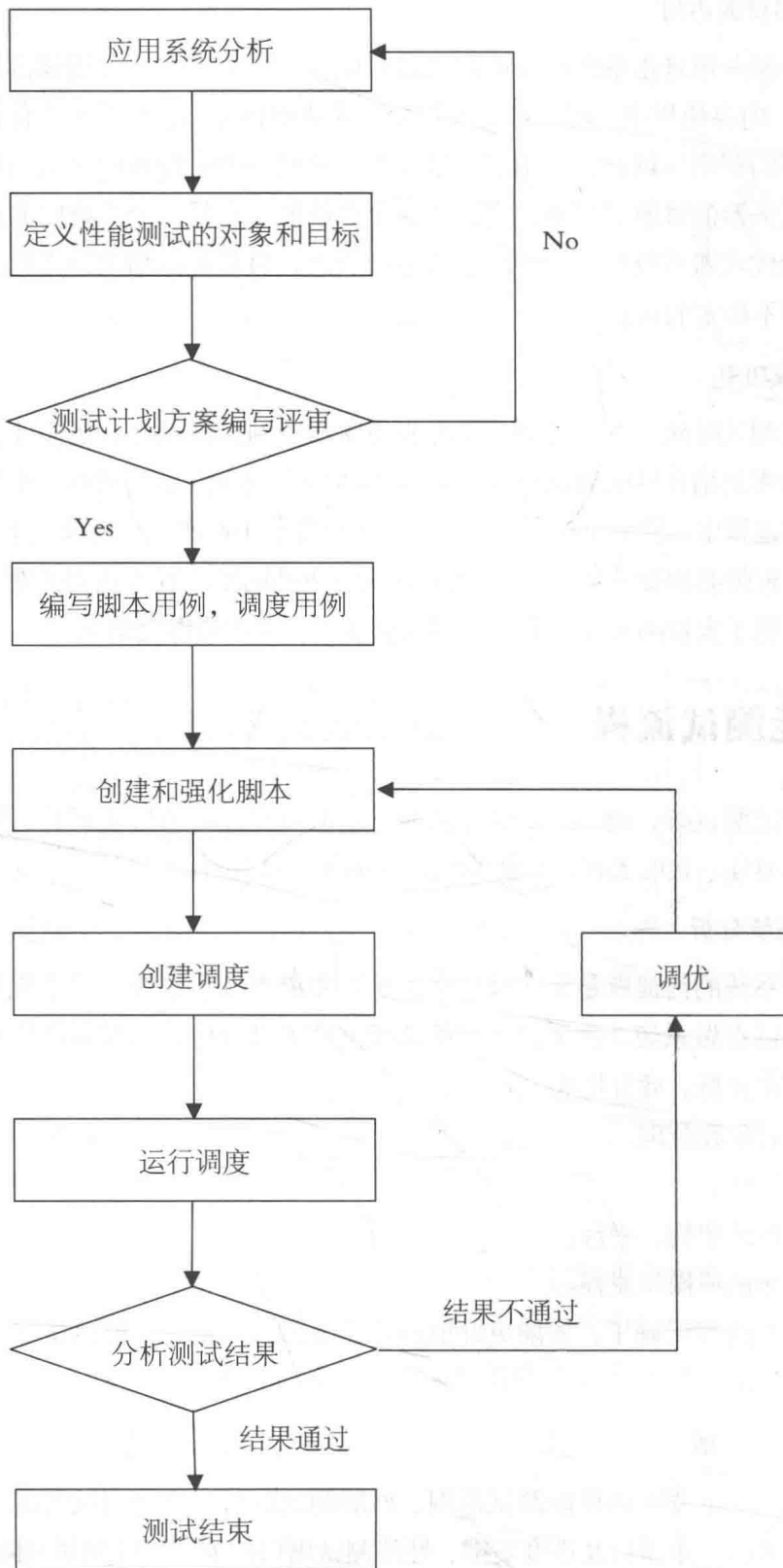


图 1-1 性能测试流程

3. 性能测试方案

性能测试方案主要是明确性能测试模型、性能测试策略、性能测试业务提取等内容。

4. 脚本用例

脚本用例主要用于指导脚本开发工程师如何开发一个性能测试脚本，应明确操作流程、开发方式（录制、手工编码、外部封装）、脚本优化方式等内容。

5. 调度用例

调度用例是用于明确调度的设置。脚本开发完成以后，要设置调度运行脚本进行测试。设置调度就是模拟出用户使用此系统的场景，比如模拟用户量、脚本运行时间长度、用户使用过程中的操作流程和波动情况。这些都是调度用例需要明确的内容。

6. 创建调度

在 RPT 中按照调度用例设置调度、运行过程中所需的监控器，为调度的运行做好准备。

7. 运行调度

调度创建完成后，开始运行调度，也就是运行性能测试。整个运行过程按照前面创建的调度进行。

8. 分析测试结果

调度运行完成以后，整个测试过程会生成大量的过程数据。测试工程师可以根据监控器监控到的数据进行结果分析。通过分析明确测试结果是否满足要求。如果满足要求，则测试结束；如果不满足要求，则需要进行调整的动作。调优后继续进行性能测试，直到性能测试结果满足要求。

1.5 性能测试的分类

根据不同的性能测试目标，性能测试的分类如下。

1. 负载测试

负载测试就是在一定的软件、硬件及网络环境下，在不同虚拟用户数量情况下，通过运行一种或多种业务测试服务器的性能指标是否在用户的要求范围内。它主要用于确定系统所能承载的最大用户数、最大有效用户数以及不同用户数下的系统响应时间及服务器的资源利用率。

负载测试强调在一定的环境下验证系统能否达到对应的指标，大多数的性能测试都是负载测试。

2. 压力测试

压力测试就是在一定的软件、硬件及网络环境下，通过模拟大量的虚拟用户向服务器产生负载，使服务器的资源处于极限状态下长时间连续运行，以测试服务器在高负载情况下能否稳定工作。压力测试强调系统的稳定性，处理能力已经不重要了。

3. 容量测试

容量测试就是在一定的软件、硬件及网络环境下，向数据库中构造不同数量级别的数据记录，通过运行一种或多种业务在一定的虚拟用户数量情况下，获取不同数据级别的服务器性能指标，以确定数据库的最佳容量。容量测试不光可以对数据库进行测试，还可以对硬件处理能力，各种服务器的连接能力等进行性能测试，目的是测试系统在不同容量级别能否达到指定的性能。

4. 配置测试

配置测试就是在不同的软件、硬件以及网络环境配置下，在一定的虚拟用户数量情况下通过运行一种或多种业务，获得不同配置的性能指标，目的是选择最佳的设备及参数配置。通过产生不同的配置，得到系统性能的变化状况。

通过配置测试可以将性能缺陷放大，方便定位性能瓶颈。

5. 基准测试

在一定的软件、硬件及网络环境下，模拟一定数量的虚拟用户运行一种或多种业务，将测试结果作为基线数据，在系统调优或者系统评测过程中，通过运行相同的业务场景并比较测试结果，确定调优是否达到效果或者为系统的选择提供决策数据。

基准测试一般基于配置测试，通过配置测试得到数据，并将这个数据作为基准，比较每次调优后的性能是否有所改善。

6. 并发测试

并发测试就是通过模拟多个用户并发访问同一个应用、同一个存储过程或数据记录以及其他并发操作，测试是否存在死锁、数据错误等故障。为了避免数据库和开发的并发错误，需要专门针对每个模块进行并发测试。

1.6 性能测试工程师技能模型

作为一个性能测试工程师，需要掌握如下知识和技能。

1. 熟悉软件测试基本理论

性能测试也是软件测试中的一种测试类型，如果不掌握软件测试的基础理论，是无法完成性能测试的。

2. 掌握软件测试常用方法

性能测试是系统测试的一种，所以必须要掌握基本的软件测试方法，例如性能测试需求分析、性能测试计划编写、性能测试方案设计、性能测试用例设计等。

3. 熟悉一门编程语言

开发性能测试脚本需要使用编程语言，如 Java 语言。如果没有编辑语言功底无法开发出好的脚本。性能测试环境的准备需要有代码功底，编写一些测试工具，完成测试环境的搭建。性能测试过程往往还需要准备大量的测试数据，需要编写一些代码才能完成测试数据的准备。在性能调试的时候需要对开发语言有一定的了解，这样才能读懂软件的核心逻辑或算法。

4. 熟悉一种数据库管理系统

数据库作为软件不可或缺的一部分，从过去的数据存储功能逐渐发展为数据库管理系统。数据库除了可以保存用户的数据，也可以通过存储过程或触发器等数据库对象完成一些业务逻辑的处理。测试工程师在做容量测试时，数据库的环境配置、数据库的问题定位和调优都要求测试工程师有很深的数据库技能背景，否则是无法胜任的。

5. 熟悉 Web 服务器

现在被测试系统大多数都是 B/S 架构系统，作为 B/S 架构 Web 服务器是不可或缺的一部分。Web 服务器会涉及被测试系统的环境搭建、环境配置、性能分析调试等内容。

6. 熟悉常见网络协议

性能测试是通过消息驱动被测系统的，消息需要符合一定的协议规则，几乎所有的性能测试工具都是基于协议来工作的。开发性能测试脚本实际就是按照协议组消息包的过程，组包以后发送给服务器。所以掌握和了解协议的原理和规则是非常必要的。另外还需要了解消息抓包工具的使用，熟悉协议分析的过程。

7. 熟练使用一种性能测试工具

性能测试需要模拟多用户执行业务流程，需要做监控。以上任务测试工程师可以通过编写性能测试工具来完成，但这样做测试工作效率比较低。业界已经提供了很多出色的性能测试工具，可满足性能测试工程师对性能工具的需求。要想顺利、高效地完成性能测试，测试工程师应掌握一个业界成熟的性能测试工具，如 RPT、Loadrunner、Jmeter 等。

1.7 性能测试工具介绍

1. IBM Rational Performance Tester

IBM Rational Performance Tester (简写 RPT) 是 IBM 基于 Eclipse 平台及开源的测试