

软件性能测试领域具有突破性创新意义的重要著作，三位资深软件测试专家多年一线工作经验结晶，业内多位测试专家联袂推荐

基于LoadRunner、Apache ab和JMeter等性能测试工具，以真实项目为依托，全面深入讲解了软件性能测试、安全测试、性能调优的流程、方法和技巧



# LoadRunner 性能测试 巧匠训练营

LoadRunner Performance Test  
From Novice to Expert

赵强 邹伟伟 任健勇 著



机械工业出版社  
China Machine Press



# LoadRunner 性能测试 巧匠训练营

LoadRunner Performance Test  
From Novice to Expert

赵强 邹伟伟 任健勇 著



机械工业出版社  
China Machine Press

## 图书在版编目 (CIP) 数据

LoadRunner 性能测试巧匠训练营 / 赵强, 邹伟伟, 任健勇著. —北京: 机械工业出版社, 2015.1  
(实战)

ISBN 978-7-111-48700-5

I. L… II. ①赵… ②邹… ③任… III. 软件工具—测试 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2014) 第 281480 号

## LoadRunner 性能测试巧匠训练营



出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 高婧雅

责任校对: 殷虹

印刷: 北京市荣盛彩色印刷有限公司

版次: 2015 年 1 月第 1 版第 1 次印刷

开本: 186mm×240mm 1/16

印张: 25

书号: ISBN 978-7-111-48700-5

定价: 69.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzjsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光/邹晓东

## Preface 前言

国内软件测试行业人才缺口已突破 20 万，并随着需求的不断攀升而挺向 30 万大关。据悉，国外成熟软件企业，一个软件开发工程师对应 1 ~ 2 个软件测试工程师，而国内软件企业，平均 8 个软件开发工程师才对应 1 个软件测试工程师，比例严重失衡。国内测试行业在这一点上与国外的差距比较大。实际上，为了保证软件质量，从项目开始测试人员就要介入，要了解客户需求，参与项目评审，把握测试要点。如果测试人员数量少，软件质量是得不到保证的。测试行业的确需要大量人才，尤其是在性能测试方面有丰富测试经验的人才更加稀缺。

另一方面，软件产业是国家鼓励发展的朝阳产业，软件产业要发展，提高软件质量势在必行，这样就产生了对软件测试工程师的大量需求。

由此可见，测试行业需求量大，前景好，但也面临激烈的竞争。所以不断提升自身技能，并明确职业发展方向对所有人来说就显得非常重要了，这样才能抓住一切机会。

### 为什么要写这本书

经常有软件测试的同行和学员向我们咨询：自己读了不少性能测试的书籍和资料，但是仍然无法把性能测试完全灵活地运用于项目中。经过深入的交流后发现，大部分都是“纸上谈兵”式的学习所造成的。对于求职者而言，花费大量精力学习的知识没法运用到项目中，十分苦恼；同样，对于公司而言，高薪招来的人员无法快速切入项目，培养成本大大增加。一本以真实的项目为核心，融技术讲解于项目实战的性能测试书籍是十分必要的。

同时我们也发现，很多书籍描述性能测试知识简单、死板，会使大部分读者感到枯燥、乏味。其实性能测试的学习是一个很快乐的过程，其中充满了挑战、惊险、失望与希望，可

以说是痛并快乐着。为了最大限度地提高阅读趣味性，强化出具体的实战测试内容（包括如何发现并解决性能问题等），更好地理解学习性能测试，本书描述了一个刚步入性能测试领域的小兵“小白”在性能测试工作中的学习、成长历程，让读者在充实而惬意、严谨又不失快慰的氛围中掌握性能测试的全部流程以及性能调优方法与技巧！（小兵成长经历纯属虚构，如有雷同，纯属巧合。）

很多读者之所以会步入性能测试领域，也是因为职业发展到了一个瓶颈期，或者是对职业发展进入了一个迷茫的状态，而市面上缺少这种类型的指导书籍。本书最后以真实的人物经历以及职业发展两个方面来帮助读者解答疑问，在黑暗中点亮一盏灯。

最后总结一下，本书并不是单纯地讲解 LoadRunner 工具，它只是书中的一部分而已。另外，写书不是为了说明自己有多牛，而是知识经验的总结、梳理与分享。不论之前大家是否了解性能，请耐心读完全书，你一定会有非常大的收获，不仅仅是技术上的。我们也相信只有更多的分享、交流才能创造更大的价值，才能为测试界的进步尽一点绵薄之力。

## 本书特色

从技术层面来讲，性能测试的复杂性是不言而喻的，目前性能测试一直处于一个只能发现问题而无法定位并给出解决方案的状态，本书正是为了改变现状而诞生的。

从内容层面来讲，目前很少有此类以全程项目实战和调优为主的性能测试类书籍，实战类型书籍向来是比较受欢迎的书籍，加之内容又以现实的故事为穿插，真实而不枯燥，可读性极强。

从读者层面来讲，很多读者以及大部分学员都没法把知识、方法和技巧应用到实际项目中，存在严重的“纸上谈兵”现象，本书将有望解决此难题。

从测试职业发展层面来讲，技术必然是重要的，但是职业的发展规划也不容忽视，能避免少走弯路，很多读者都对测试的发展有疑问，本书中主人公小白的螺旋上升式发展和众多从业者实录会对读者有所启迪。

## 读者对象

不论你是否有性能测试基础，也不论你是否了解性能测试，抑或是你已经是性能测试老手，本书都会对你有所帮助。包括但不限于以下读者：

□ 测试工程师

- 开发工程师
- 运维工程师
- 测试经理
- 技术主管
- 对性能测试感兴趣的读者

## 如何阅读本书

本书分为四大部分：

第一篇（第 1 ~ 7 章）为“初入职场”篇，全面、细致地讲解了刚刚步入性能测试领域的小白是怎么一步步学习性能测试的，并以实战方式讲解所有知识点。

**第 1 章** 本章用实际例子深入浅出地来讲解性能测试的基本知识与模型，带领读者快速入门，摆脱枯燥。

**第 2 章** 本章对 LoadRunner 的脚本开发进行详细实战讲解，并对重要知识点进行深入分析总结。

**第 3 章** 本章以 LoadRunner 的 Controller 为引子，深入讲解实战中的场景设计、资源监控等重点知识。

**第 4 章** 本章以真实案例为蓝本，通过 LoadRunner 的 Analysis 进行深入分析，并得出性能测试报告。

**第 5 章** 介绍了开源性能测试工具 JMeter 的基本使用，并结合 HTTP、JDBC、SAOP 的实际例子演示了 JMeter 参数化、关联、检查点的基本使用方法。

**第 6 章** 本章简单介绍了开源性能测试工具 Apache ab 的基本使用方法，方便大家开阔视野。

**第 7 章** 本章主要对常用的脚本协议开发进行了全面、深入的实战讲解，以具体的例子演示实际应用。

第二篇（第 8 章）为“新硌初试”篇，重点以实战方式讲解在 .NET 项目中如何进行性能测试以及性能调优，并把所有知识点融合到项目中，使读者更好地理解并应用，摆脱无趣的理论。

**第 8 章** 本章以 .NET 项目的案例为基础，全面讲解了完整的性能测试项目是如何进行的，把以往的知识点再次融入到实战中。

第三篇（第 9 章、第 10 章）为“再露锋芒”篇，重点以实战方式讲解 Java 项目的性能

测试，通过真实的、完整的项目使读者身临其境地体会性能测试、性能调优、安全测试以及最佳流程实践等。

**第9章** 本章用一个实际的项目完整地介绍了真正的前沿互联网公司项目实战是怎么做的，用真实例子带领大家了解真正前沿的性能测试思想。

**第10章** 本章全面讲解了安全测试的原理，并结合常用的安全测试工具进行了实战演示。

第四篇（第11章、第12章）为“反躬自省”篇，以真实的在职人物的学习历程、心得以及方法做交流实录，再次以事实指导读者，并针对现状来分析测试人员的职业发展，指导读者的选择与提升路线。

**第11章** 本章讲述了真实人物在面对转型、提升过程中的奋斗历程，以最淳朴的思想感动你我。

**第12章** 本章从职业发展角度全面剖析测试行业的点点滴滴，为所有测试人员解惑并给出指导建议。

其中第二、三篇以实战项目来讲解性能测试的应用，如果你是一名有一定基础的性能测试工程师，那么你可以直接阅读这部分内容。但是如果你是一名初学者，请一定从第一部分开始学习。如果你目前对自己的工作有迷茫之处或想得到职业发展上的指导，可看第四篇。本书为读者提供了完整的流程以及细致的解决方案，也能看到小白的成长与不足，这也许正是你的真实写照。

附录总结了 LoadRunner 常见的问题以及解决方法，让你不必再四处求助。同时也为对编写报告文档感到很头疼的读者提供了各类参考模板。

## 勘误和支持

由于笔者的水平有限，编写时间仓促，书中难免会出现一些错误或者不准确的地方，恳请读者批评指正。你可以将书中的错误发布在 BestTest 网站（<http://www.besttest.cn>），同时如果你遇到任何问题，也可以加入我们的 QQ 群：249854093，我们将尽量在线上为读者提供最满意的解答。如果你有更多的宝贵意见，也欢迎发送邮件至 [besttest@foxmail.com](mailto:besttest@foxmail.com)，期待能够得到你们的真挚反馈。

## 致谢

本书由赵强、邹伟伟、任健勇撰写，其中，赵强写了第1、2、3、4、7、8、11、12章；邹伟伟写了第5、6、9章；任健勇写了第10章。

感谢广大读者对三个老男人的支持，你们的支持与热情才是我们写这本书的最大动力。

感谢机械工业出版社华章公司的杨福川和高婧雅，在这段时间中始终支持我们的写作，你们的鼓励和帮助使得我们能顺利完成全部书稿。

特别感谢 BestTest 的所有学员给我们提供了不少宝贵的建议与意见，使得本书更加通俗易懂且有趣味性。

最后，谨以此书献给支持我们的小伙伴们以及广大测试同仁，谢谢你们的一路支持与伴随！

赵强 邹伟伟 任健勇

# 目 录 Contents

## 前言

## 第一篇 初入职场

### 第1章 与性能测试的亲密触碰 ..... 2

#### 1.1 性能测试的作用以及重要性 ..... 2

##### 1.1.1 由性能引发的严重问题 ..... 2

##### 1.1.2 性能测试的重要性以及 必要性 ..... 3

##### 1.1.3 什么系统需要做性能测试 ..... 4

##### 1.1.4 性能测试的目的 ..... 4

#### 1.2 生活中的性能测试 ..... 5

#### 1.3 性能术语与指标详解 ..... 5

#### 1.4 性能测试分类详解 ..... 9

#### 1.5 性能测试模型分析 ..... 11

##### 1.5.1 曲线拐点模型分析 ..... 12

##### 1.5.2 地铁模型分析 ..... 13

#### 1.6 本章小结 ..... 14

### 第2章 LoadRunner VuGen 实战 ..... 15

#### 2.1 LoadRunner 快速安装 ..... 15

#### 2.2 LoadRunner 组成与示例程序 ..... 16

#### 2.3 深入理解协议选择 ..... 18

#### 2.4 脚本编写实战 ..... 19

##### 2.4.1 脚本录制 ..... 20

##### 2.4.2 “利器”——函数手册 ..... 22

##### 2.4.3 脚本优化 ..... 23

##### 2.4.4 脚本回放 ..... 23

#### 2.5 脚本增强实战 ..... 24

##### 2.5.1 检查点实战 ..... 24

##### 2.5.2 参数化实战 ..... 26

##### 2.5.3 关联实战 ..... 34

##### 2.5.4 逻辑判断增强实战 ..... 38

##### 2.5.5 事务实战 ..... 40

#### 2.6 脚本调试实战 ..... 42

##### 2.6.1 message 函数详解 ..... 42

##### 2.6.2 断点的应用 ..... 42

##### 2.6.3 日志的查看分析 ..... 43

##### 2.6.4 自带浏览器的使用 ..... 44

#### 2.7 VuGen 使用技巧 ..... 44

##### 2.7.1 协议嗅探 ..... 45

##### 2.7.2 通用设置 ..... 45

##### 2.7.3 录制设置 ..... 46

##### 2.7.4 Run-time Settings 设置 ..... 48

#### 2.8 本章小结 ..... 52

### 第3章 LoadRunner Controller

#### 实战 ..... 53

#### 3.1 Controller 介绍 ..... 53

#### 3.2 场景设计实战 ..... 53

##### 3.2.1 集合点实战 ..... 55

##### 3.2.2 IP 欺骗实战 ..... 56

##### 3.2.3 添加 Windows 和 Linux 压力机 实战 ..... 59

##### 3.2.4 场景设计实战 ..... 61

#### 3.3 场景监控实战 ..... 62

##### 3.3.1 监控信息概览与自定义 脚本 ..... 62

##### 3.3.2 Windows 和 Linux 的监控 以及指标讲解 ..... 65

#### 3.4 本章小结 ..... 69

### 第4章 LoadRunner Analysis

#### 实战 ..... 70

#### 4.1 Analysis 介绍 ..... 70

#### 4.2 性能数据图表分析实战 ..... 72

##### 4.2.1 概要图分析 ..... 72

##### 4.2.2 虚拟用户图分析 ..... 73

##### 4.2.3 错误图分析 ..... 75

##### 4.2.4 事务图分析 ..... 77

##### 4.2.5 Web 资源图分析 ..... 80

##### 4.2.6 网页细分图分析 ..... 85

##### 4.2.7 系统资源图分析 ..... 88

#### 4.3 图表的合并与关联 ..... 89

#### 4.4 数据图表的设置 ..... 90

#### 4.5 性能测试报告实战 ..... 92

##### 4.5.1 性能测试报告编写技巧 ..... 92

##### 4.5.2 性能测试分析思路 ..... 93

##### 4.5.3 性能测试报告编写 ..... 94

#### 4.6 本章小结 ..... 96

### 第5章 开源性能测试工具 Apache

#### JMeter 快速入门 ..... 97

#### 5.1 Apache JMeter 概述 ..... 97

#### 5.2 JMeter 的安装与使用 ..... 99

#### 5.3 JMeter 快速入门之 HTTP 脚本 实战 ..... 101

##### 5.3.1 测试计划 ..... 101

##### 5.3.2 Vuser 及场景设计 ..... 102

##### 5.3.3 JMeter 之脚本录制 ..... 103

##### 5.3.4 JMeter 参数化 ..... 107

##### 5.3.5 JMeter 关联 ..... 111

##### 5.3.6 JMeter 检查点 ..... 114

##### 5.3.7 思考时间和集合点 ..... 115

##### 5.3.8 运行结果分析 ..... 117

##### 5.3.9 JMeter 之分布式测试 ..... 120

#### 5.4 JMeter JDBC 脚本实战 ..... 121

#### 5.5 JMeter SOAP 脚本实战 ..... 124

#### 5.6 JMeter 使用注意事项 ..... 127

#### 5.7 本章小结 ..... 128

### 第6章 开源性能测试工具 Apache

#### ab 快速入门 ..... 129

#### 6.1 Apache ab 安装使用 ..... 129

#### 6.2 Apache ab 快速入门 ..... 129

#### 6.3 Apache ab 实战 ..... 130

#### 6.4 Apache ab 总结注意事项 ..... 132

#### 6.5 本章小结 ..... 133

|                                     |     |
|-------------------------------------|-----|
| <b>第7章 最佳脚本开发实战</b>                 | 134 |
| 7.1 接口测试脚本开发实战                      | 134 |
| 7.1.1 GET 接口实战                      | 134 |
| 7.1.2 POST 接口实战                     | 135 |
| 7.1.3 使用 LoadRunner 完成接口<br>功能自动化测试 | 136 |
| 7.2 常用协议脚本开发实战                      | 138 |
| 7.2.1 Web Service 协议脚本开发<br>实战      | 139 |
| 7.2.2 ODBC 协议脚本开发实战                 | 141 |
| 7.2.3 Java 协议脚本开发实战                 | 142 |
| 7.2.4 Windows Sockets 协议脚本<br>开发实战  | 143 |
| 7.2.5 移动应用协议脚本开发<br>实战              | 145 |
| 7.2.6 手工编写 FTP 脚本                   | 147 |
| 7.3 使用 LoadRunner 打点监控<br>Tomcat    | 148 |
| 7.4 本章小结                            | 150 |

## 第二篇 新研初试

### 第8章 .NET 项目性能测试全程

#### 实战

|                 |     |
|-----------------|-----|
| 8.1 项目背景与架构分析   | 153 |
| 8.2 测试环境需求确认与搭建 | 155 |
| 8.2.1 测试环境需求确认  | 155 |
| 8.2.2 测试环境搭建    | 155 |
| 8.3 性能测试工具选型    | 158 |
| 8.4 业务建模与用例设计   | 159 |

|                 |     |
|-----------------|-----|
| 8.4.1 业务场景分析    | 159 |
| 8.4.2 性能需求分析与提取 | 161 |
| 8.4.3 性能测试用例设计  | 162 |
| 8.5 脚本开发与优化     | 164 |
| 8.6 执行测试        | 168 |
| 8.6.1 场景设置      | 168 |
| 8.6.2 监控设置      | 171 |
| 8.6.3 运行场景与实时监控 | 173 |
| 8.7 性能测试分析与调优建议 | 177 |
| 8.7.1 性能结果分析    | 178 |
| 8.7.2 性能调优建议    | 184 |
| 8.8 本章小结        | 190 |

## 第三篇 再露锋芒

### 第9章 Java 项目性能测试全程

#### 实战

|                        |     |
|------------------------|-----|
| 9.1 项目目标分析             | 192 |
| 9.1.1 系统架构             | 193 |
| 9.1.2 业务流程             | 194 |
| 9.2 性能测试需求获取以及确定       | 197 |
| 9.2.1 性能测试环境需求描述       | 197 |
| 9.2.2 性能测试环境需求获取       | 197 |
| 9.2.3 性能测试需求确定         | 202 |
| 9.3 性能测试场景获取以及用例<br>设计 | 204 |
| 9.3.1 性能测试场景选取与制定      | 205 |
| 9.3.2 性能测试数据确定         | 207 |
| 9.3.3 性能测试用例设计         | 208 |
| 9.4 性能测试环境准备与搭建        | 216 |

|        |                        |     |
|--------|------------------------|-----|
| 9.4.1  | 性能测试环境                 | 216 |
| 9.4.2  | 性能测试环境的重要性             | 217 |
| 9.4.3  | 确定实际运行环境               | 218 |
| 9.4.4  | 申请并搭建仿真性能测试环境          | 219 |
| 9.4.5  | Mock Server 的准备        | 220 |
| 9.5    | 操作系统性能监控分析工具的选择与使用     | 222 |
| 9.5.1  | Windows 性能监控工具选择以及监控详解 | 222 |
| 9.5.2  | Linux 监控工具选择以及监控详解     | 234 |
| 9.6    | 数据库系统性能监控工具选择以及使用      | 254 |
| 9.6.1  | Oracle 监控分析            | 254 |
| 9.6.2  | MySQL 监控分析             | 264 |
| 9.7    | 中间件性能监控工具以及使用          | 267 |
| 9.8    | JVM 性能监控工具选择以及使用       | 268 |
| 9.8.1  | JVM 内存结构以及内存管理基础知识     | 268 |
| 9.8.2  | JVM 性能监控分析调优           | 273 |
| 9.9    | 性能测试数据准备与制作            | 297 |
| 9.10   | 性能测试脚本开发与优化            | 300 |
| 9.11   | 性能测试执行与监控              | 308 |
| 9.11.1 | 场景设置                   | 308 |
| 9.11.2 | 监控设置                   | 310 |
| 9.11.3 | 项目性能问题的发现、定位与分析        | 310 |
| 9.12   | 性能回归测试与结果              | 321 |
| 9.12.1 | 回归测试后的性能测试结果           | 322 |

|               |                 |     |
|---------------|-----------------|-----|
| 9.12.2        | 数据库性能优化建议       | 323 |
| 9.13          | 本章小结            | 326 |
| <b>第 10 章</b> | <b>安全测试</b>     | 327 |
| 10.1          | 偶遇 Web 安全测试     | 327 |
| 10.1.1        | Web 安全测试定义      | 328 |
| 10.1.2        | Web 安全测试流程      | 329 |
| 10.2          | Burp Suite 使用浅析 | 330 |
| 10.3          | 安全测试原理与实战       | 337 |
| 10.3.1        | 绕过客户端漏洞         | 338 |
| 10.3.2        | 攻击验证机制          | 341 |
| 10.3.3        | 攻击会话管理          | 348 |
| 10.3.4        | SQL 注入攻击        | 353 |
| 10.3.5        | XSS 攻击          | 355 |
| 10.3.6        | CSRF 攻击         | 358 |
| 10.4          | 本章小结            | 360 |

## 第四篇 反躬自省

|               |                      |     |
|---------------|----------------------|-----|
| <b>第 11 章</b> | <b>测试工程师转型实录</b>     | 362 |
| 11.1          | 从非 IT 业到性能测试工程师的华丽转型 | 362 |
| 11.2          | 好妈妈转型性能测试，年龄不是阻碍     | 363 |
| 11.3          | 从功能测试到性能测试的完美转型      | 364 |
| 11.4          | 本章小结                 | 365 |
| <b>第 12 章</b> | <b>测试工程师职业发展分析</b>   | 366 |
| 12.1          | 软件测试现状与发展前景          | 366 |

|      |                    |     |
|------|--------------------|-----|
| 12.2 | 如何成为测试工程师 .....    | 367 |
| 12.3 | 测试工程师的实际价值 .....   | 368 |
| 12.4 | 测试工程师的危机 .....     | 370 |
| 12.5 | 测试工程师的职业发展方向 ..... | 371 |
| 12.6 | 本章小结 .....         | 372 |

|             |                                    |            |
|-------------|------------------------------------|------------|
| <b>附录 A</b> | <b>LoadRunner 常见问题以及解决方法 .....</b> | <b>373</b> |
|-------------|------------------------------------|------------|

|             |                         |            |
|-------------|-------------------------|------------|
| <b>附录 B</b> | <b>常用性能测试文档模板 .....</b> | <b>376</b> |
|-------------|-------------------------|------------|

|             |                      |            |
|-------------|----------------------|------------|
| <b>附录 C</b> | <b>正则表达式全集 .....</b> | <b>377</b> |
|-------------|----------------------|------------|

|             |                              |            |
|-------------|------------------------------|------------|
| <b>附录 D</b> | <b>MySQL 数据库事务隔离级别 .....</b> | <b>380</b> |
|-------------|------------------------------|------------|

|             |                              |            |
|-------------|------------------------------|------------|
| <b>附录 E</b> | <b>Perfmon 计数器参数说明 .....</b> | <b>382</b> |
|-------------|------------------------------|------------|

|             |                       |            |
|-------------|-----------------------|------------|
| <b>附录 F</b> | <b>学习资料网址摘录 .....</b> | <b>387</b> |
|-------------|-----------------------|------------|



## 第一篇 Part 1

# 初入职场的

小白多年一直勤勤恳恳地奋斗在测试行业，但始终每天重复着黑盒功能测试，虽然自学过一些其他测试技能，但总感觉不系统，自己也深感职业发展到了瓶颈期，希望能有所突破，于是他开始查阅资料，无意间发现了BestTest这样一个网站，里面有不少好资料，经过一段时间的考虑，决定向性能测试的方向发起突击，希望早日突破自己的瓶颈！

说来也巧，这时候收到经理的一封邮件，内容如下。

亲爱的小白：

随着公司与产品的发展，我们的测试技术与手段也要与时俱进，所以准备在后续项目中增加性能测试，而这方面我们没有技术储备，希望你能承担起这个任务，抓紧时间学习性能测试，争取早日应用到我们的项目里。不知你是否愿意接受这个挑战？

这么好的机会小白怎能放过，于是小白毅然决然地接受任务并信心满满地开始了性能测试学习之旅。

.....

# 与性能测试的亲密触碰

性能测试的挑战性和趣味性小白早有耳闻，也会经常听到各个公司因为系统性能而引发的一系列严重问题，所以性能测试会越来越受到重视，只是时间的问题。下面就让我们和小白先来了解下性能测试的趣事，再一同学习性能测试的基本知识。

## 1.1 性能测试的作用以及重要性

随着社会的发展，用户对产品的要求也越来越高，以前可能看重功能方面，现在正在逐步转变为性能方面，同时各大公司也加强了产品的性能测试，因为从这几年发生的事件来看，性能带来的严重问题以及损失不容忽视，而性能测试的重要性也不言而喻。

### 1.1.1 由性能引发的严重问题

小白印象中由性能引发的严重问题历历在目，大部分都是由于没有做性能测试、性能测试做得不充足或者对并发以及流量的预估不正确导致。

**【案例 1】**2008 年的奥运会票务系统，由于庞大的订票人数超出预期，奥运票务系统“开工”后不久便陷入“瘫痪”状态，当时对外公布的是奥运票务系统每小时能处理 15 万张门票的销售，以及承担每小时 100 万次以上的网上浏览量，但 10 月 30 日系统死机前每小时的网上浏览量达到 800 万，1 小时售出的票也达到了 20 万张。由于预估工作的缺陷，导致很多人无法通过网络订到自己想要的票，影响了很多人的热情，也损害了国家形象。

**【案例 2】**作为电商的代表，2009 年 11 月 22 日，eBay 网站出现死机，导致卖家至少损失了当日销售额的 80%，原因是那年圣诞临近时，eBay 网站上有超过 2 亿件待售商品，这

个数字比上一年同期多出 33%，正是这激增的 33% 的待售商品导致 eBay 网站不堪重负而死机，80% 的销售额对于 eBay 来说不可谓不严重。

【案例 3】魔兽世界在中国的代理商由九城变更为网易，与九城服务器经常死机不无关系，但是换作网易后，服务器也经常死机。2010 年 10 月 11 日，魔兽世界服务器故障时，官网论坛上的游戏玩家纷纷发“贺词”表示不满，从这可以看出网易公司对魔兽世界的性能预估存在不足。也正是因为对性能严重忽视间接导致了九城在失去魔兽世界之后，从一家土豪公司成了一家几乎被人遗忘的公司。

【案例 4】视频网站优酷网也在 2010 年发生死机事件，超过 3 小时无法访问。优酷对外宣称的原因是：此次死机事件起源于“地球一小时”活动，优酷网为响应这次活动，全站采用关灯模式，意在借此提醒网民注重环保与节约。但此举令网友一时无法适应，大量网友频繁刷新页面导致优酷网服务器崩溃。

【案例 5】2010 年，中国最大的微博平台新浪微博死机 4 小时，新浪官方解释说：之所以掉线几小时，是因为用户增长超出预期，服务器备感压力。自上午 10 点起，用户无法登录，新浪的报错页面几次更改，最初的“微博正在升级，将于 11:30 恢复”，然后改为“12:00 恢复”，过了一段时间，干脆改为“稍后恢复”，然而，估计是看不到恢复希望，提示信息又改为“微博系统压力过大正在抢修，我们深表歉意”。悲剧的是“歉意”竟然写成了“谦意”，这件事遭到网友的大量恶搞，小白也是参与者之一。

### 1.1.2 性能测试的重要性以及必要性

根据 2008 年 Aberdeen Group 的研究报告表明，Web 网站 1s 的页面加载延迟相当于少了 11% 的 PV，相当于降低了 16% 的顾客满意度。如果从金钱的角度计算，就意味着：如果一个网站每天挣 10 万元，那么一年下来，由于页面加载速度比竞争对手慢 1s，可能导致总共损失 25 万元的销售额。

Compuware 公司分析了超过 150 个网站和 150 万个浏览页面，发现页面响应时间从 2s 增长到 10s，会导致 38% 的页面浏览放弃率。

Radware 也曾发布一份题为“行业现状：2013 年春季电商页面速度与 Web 性能”的调查报告。报告指出，仅一年时间内，美国前 2000 家领先的在线零售商网站的加载时间较去年同期减慢了 22%，网站性能急剧下降，用户体验质量大幅下降。对网站回访率、跳出率、客户满意度及在线收入等多个关键业务指标的影响越来越大，对在线零售商而言，网站加载速度已经成为制约其发展的重要因素，提升 Web 性能已经刻不容缓。

2014 年中国电子商务研究中心发布对电商网站的调查报告，报告中指出用户对网站响应时间的要求很严苛，期望立刻做出响应的占 90%，期望 5min 内做出响应的占 10%。

从上面的研究分析再结合小白印象中列举的例子可以看出，性能测试非常重要也非常必要，因为性能问题不仅仅会损害公司的形象，也会造成公司资金方面的损失。

### 1.1.3 什么系统需要做性能测试

小白刚接触到“什么系统需要做性能测试”这个问题的时候，心里在想应该是大型的系统、软件才需要做性能测试，如果只是几个人用就没有必要了。可仔细想想，小白觉得应该是所有系统、软件都应该做性能测试，关键是要思考应该做到什么程度，而不是做不做的问题。因为就算是一个人在使用某个系统，但该系统的查询性能极差，一次查询需要 50 多秒钟，这绝对是任何人都难以接受的。

接着小白对现有的系统进行了分类，大致分为单机系统、C/S、B/S。这 3 类系统都应该进行性能测试，只是每个分类有各自特点，在实际测试中应该有不同策略进行应对。

一般 C/S 架构的应用程序更关注于系统资源使用情况、数据库性能以及运行的配置要求等。例如，内存、用户连接数、数据库死锁、数据库 cache 命中率、运行的最低配置等。

而对于 B/S 架构的应用程序，会关注 Web 服务器的相关指标，如每秒点击数、吞吐量、尝试连接数、事务成功率等。

如下几个案例分别针对典型的系统进行了说明。

【案例 1】假设使用 Word 来编辑一个 1 000 多页的文档，该文档包含了丰富的图表、图片，需要等待系统花多少秒的时间进行处理。这时需要关注性能响应。

【案例 2】某业务系统属于二次开发，之前没有做过性能测试，当并发 100 个用户时就会造成数据库服务器崩溃。这是很明显的性能问题。

【案例 3】某企业内部信息系统，使用人比较少，但并发时会出现重复的相同记录。这种场景很难在功能测试时出现，所以有时候性能测试并不是只能发现性能问题。

【案例 4】面向广大互联网人群的网站，每天都需要接受大量的访问请求，服务器压力大，对这样的系统进行性能测试是十分必要的。

其中 B/S 架构的系统会比较复杂，小白接到的正好是 B/S 的项目，看来这下需要学习一番了。

### 1.1.4 性能测试的目的

很多第一次接触性能测试的人都会把功能测试的思想带入，造成思维的局限。其实性能测试还是与功能测试有所不同的。性能测试更加关注系统的性能表现，也就是 How fast 和 How much。而做性能测试就是排除系统瓶颈，使得它表现得更好、更霸气。可以从以下几个方面来理解。

1) 评估当前系统。系统未做过任何性能测试，对系统的当前性能情况不了解，就好像没有体检过就不知道自己的身体状况一样。而此前说到的一系列性能引发的严重问题也正是由于缺少了必要的性能评估而导致。

2) 寻找瓶颈，优化性能。常见的现象为，某业务操作响应时间很长、某系统上线一段时间后运行越来越慢，这些都需要逐步分析定位并调优。

3) 预测未来性能。当用户数和业务量增加时能否及时应对？如何调整？是增加应用服