



Windows

软件测试探秘

王磊 韩静 等著

- 自动化测试、性能测试、稳定性测试、兼容性测试、产品评测等在Windows应用程序测试中显得举足轻重，往往决定了一款客户端软件的更加宏观的质量状态；
- 敏捷、持续集成的引入，提高了客户端软件的开发与测试的效率，缩短了客户端软件的发布周期，使得客户端软件在互联网时代背景下，也能快速满足用户需求。



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

Windows 软件测试探秘

王磊 韩静 等著



电子工业出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书是百度质量部几位测试工程师以实际项目为依据,结合技术改进,讲解在大型公司中测试人员工作的主要内容,让读者从中了解具体测试工作的实际情况,从中受到启发,为自己的工作带来变化。

就 Windows 应用程序测试而言,本书除了阐述传统测试领域中的自动化测试、性能测试、稳定性测试外,还深入讲述了不同用户环境测试的方方面面:诸如兼容性测试、产品评测,以及用户界面功能测试的设计和执行情况。本书使用的案例介绍都来源于实际项目,适合于从事 Windows 客户端测试的同行及即将进入测试行业的大中专院校的毕业生阅读。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

Windows 软件测试探秘 / 王磊等著. —北京:电子工业出版社, 2013.1
ISBN 978-7-121-19157-2

I. ①W… II. ①王… III. ①Windows 操作系统—程序设计 IV. ①TP316.7

中国版本图书馆 CIP 数据核字(2012)第 291037 号

责任编辑:高洪霞

印 刷:北京东光印刷厂

装 订:三河市鹏成印业有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×980 1/16 印张:15.25 字数:330 千字

印 次:2013 年 1 月第 1 次印刷

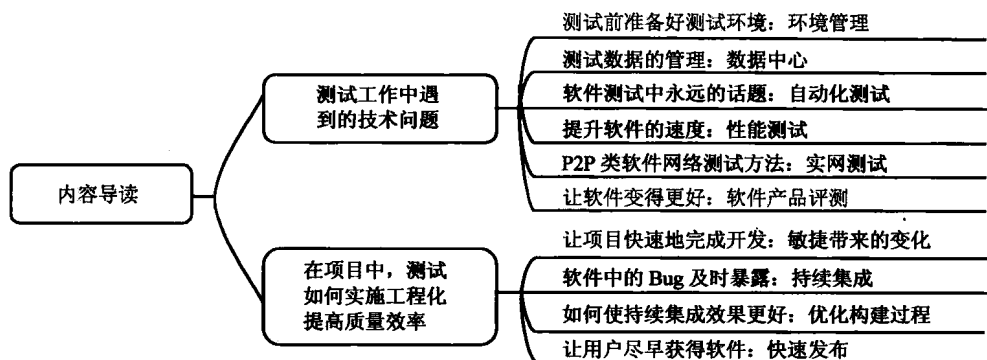
印 数:4000 册 定价:35.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前言



为什么要写这本书

随着业界对软件测试的重视，关于软件测试方面的书籍也逐渐增多。目前软件测试书籍的内容也挺广泛，常见的有测试设计、测试管理、测试团队组建、测试理论、测试过程、测试脚本编写、自动化测试、性能测试等，面面俱到。但是涉及实际的工程，测试如何进行、测试工程师如何进行实际的工作方面的书籍却不多。而针对工作中，测试工程师如何思考问题、如何改进测试方法从而提高软件的质量和开发效率的书就更少了。

于是，我们想要写一本关于测试工程师在企业中如何工作、会做哪些方面工作的书。同时也希望把我们测试工作中的一些经验与读者分享，使读者了解在实际项目中，测试

工作是如何进行的。当然我们的工作不是尽善尽美的，肯定会有更好的办法出现，希望读者能和我们交流，讨论任何的测试技术问题，共同提高测试的技术水平。

本书是“百度测试系列”的第一本，该系列的其他书籍将陆续面世，敬请读者关注。

本书讲什么

Windows 应用程序贴近于用户，伴随着互联网时代的到来，更是充当着通往整个互联网的入口，其本身的特性就决定了对于 Windows 应用程序的测试，需要面对更加复杂的用户软硬件环境和复杂网络环境，测试所需要关注的信息和内容更加丰富，这给 Windows 应用程序测试带来了巨大的挑战，同时也给 Windows 应用程序测试工作带来了更大的乐趣。

像传统测试领域中的自动化测试、性能测试、稳定性测试一样，在 Windows 应用程序测试范畴内，除了需要解决基本的技术问题外，还需要关注不同用户环境所带来的影响；诸如兼容性测试、产品评测等在服务器产品中关注度比较低的测试工作，在 Windows 应用程序测试中显得举足轻重，往往决定了一款客户端产品宏观的质量状态；对于以用户界面（UI）为交互接口的产品来说，其功能测试的设计和执行方法，更是区别于服务器产品的测试，有更大的空间和组合场景等待测试人员去探索 and 发现。

在敏捷测试模式逐渐被认可和应用的年代，Windows 应用程序的测试和开发工作结合得更加紧密，产品的质量不再仅依靠测试人员保证，开发人员也起了非常重要的作用。在整个研发体系中，质量保证工作也会像产品架构一样，分层次进行，测试作为一项服务对产品质量进行贡献，不受限于具体的工作职责划分：类似单元测试、可测性架构支持、性能瓶颈分析等工作开始贯穿到整个研发过程，而在此过程中对测试人员的要求会逐渐提升，往往会以专家和顾问的角色出现在产品研发过程中，主导整个产品质量保证工作。

本书所介绍的一系列 Windows 应用程序测试技术和方法都是基于敏捷研发模式框架，在实际项目中积累起来的，相对于传统测试理论的书籍更加具有实战参考价值，是项目实战中成功或失败经验的升华和结晶。

本书的读者

以下几个方面的读者可能会受益：

- 从事 Windows 客户端测试的同行；
- 喜欢接受测试技术挑战的朋友；
- 即将进入测试行业的院校毕业生；
- 想了解百度如何测试的测试爱好者。

本书的内容

本书是一本技术性的书籍，主要介绍技术内容，流程性的内容涉及得比较少。全书分为两个部分。上半部分为技术篇，主要介绍测试用到的相关技术，通过技术手段使测试做得更高效，测试的质量更高。下半部分为工程篇，主要介绍工程实际测试项目如何实施持续集成及通过持续集成测试做了哪些改变。

上篇：技术篇

第 1 章 测试环境管理

测试环境在测试中起着很重要的作用，一次测试的成功与否，测试环境的正确与否占了很大比重。然而繁多的 Windows 操作系统、层出不穷的各种应用软件，使得部署、维护测试环境包含了太多的重复性劳动，以及硬件资源的严重浪费。本章详细介绍如何能够在短时间内全自动化部署一个用户所需的 Windows 测试环境，如何在团队之间可视化地高效共享已有测试环境，以及如何高效地利用机器资源池。

第 2 章 测试数据中心

测试数据对于测试工作来说至关重要，特别是对于偏数据处理类的产品来说更是如此。测试数据中心的作用除了普遍能想到的管理各种测试数据的功能外，还需要考虑日常测试工作中的实际测试需求，特别是建立从测试数据到被测对象之间的“通信链路”，以便将大批量的测试数据导入到被测对象中。本章从测试数据中心的一般性需求入手，介绍期望的抽象平台模型及相应的实例参考，为需要进行大规模测试数据管理和测试辅助执行的客户端类产品提供方法和技术上的支持。

第3章 自动化测试

自动化测试始终是测试领域被关注的焦点，其涉及的方面很多：根据被测对象的不同，所需要的自动化支撑技术也会有所区别，本章将从常见的 PC 客户端 UI 自动化驱动和识别技术讲起，对于工程实践中各种技术方案的特点和应用场景进行分析，为读者打开进行自动化测试工作所需要的第一步技术积累。除此之外，也会从工程实践角度——为读者呈现自动化测试领域中经常涉及的用例开发和组织模式、框架管理等常见环节。本章所涉及的自动化测试技术主要面向客户端产品的系统级 UI 自动化，希望大家通过本章阅读开启对自动化技术和工程实践的认识和理解。

第4章 性能测试

软件的性能直接决定了用户在使用过程中的体验，而如何衡量客户端软件的性能、不同客户端软件之间的性能如何比较，则是性能测试工程师必须回答的问题。在本章的前4节中，介绍关于客户端性能测试的指标、环境和测试的方法，以及如何进行测试场景的设计。在本章最后一节，还结合具体的项目，将本章前几节中所描述的内容拼接起来，让读者看到这些理论知识是如何在项目中应用的。

第5章 实网测试

网络带宽成本大幅降低，用户的网络应用需求与日俱增，各种依赖互联网的客户端软件日益增多。这类应用程序的测试往往需要实验室模拟用户网络环境，然而网络环境模拟的高技术门槛、硬件资源成本及用户繁多的网络环境，极大降低了实验室测试这类软件的质量和效率。本章介绍如何高效、低成本地测试网络应用软件。

第6章 评测

产品评测是测试工作的扩展和延伸，在传统测试领域中，多数测试工作的目标是发现 Bug，但在评估标准上有严格的定义。对于评测来说，目标往往是测试或者评估产品的宏观质量，例如输入法的词库质量、搜索结果的排序质量等，很难用 Bug 来描述产品的这类宏观指标。本章就对这一类特殊的“测试”工作进行介绍，包括评测工作所需要考虑的各个环节，如评测指标定义、评测数据、评测环境、评测对象、评测结果分析等一整套系统分析过程。

下篇：工程篇

- 第 7 章** 敏捷带来的改变 当传统的开发模式已经不能满足现在对软件快速发布、快速迭代的要求时，引入敏捷方式势在必行。然而实施敏捷获得成功并取得收益的团队却寥寥无几，大多数团队缺少专业的敏捷教练，这使得他们在敏捷的道路上走了不少弯路。本章与读者分享在敏捷团队建设和项目管理方面的一些亲身经验和教训，所有信息均来自真实项目的实践活动，而非理论。读者可以了解在一个产品项目中如何做迭代规划、需求管理、计划跟踪及信息传递，具体到如何确定一个适合的迭代周期，如何写好一个用户故事，如何增加信息的透明度和一致性，并提供了一些项目管理文档模板，供读者借鉴使用。
- 第 8 章** 持续集成 持续集成也是近年谈论得比较热的话题，讨论在工程实践上需要做哪些基础技术工作，才能使持续集成起到良好的作用，并在工程开发中提高效率。本章介绍持续集成的各个组成部分、各个组成部分是如何实现的、它们的技术难点，以及一些关键解决方案。
- 第 9 章** 构建过程优化 部署起持续集成之后并开始按照持续集成要求运转，这时候只能得到 60 分，为了能让持续集成中的各个环节发挥出完全的功效，还需要对其中很多方面做精雕细刻。本章介绍持续集成做到一定程度之后遇到的各种问题，以及这些问题的优化方法和解决方案。本章使用实例来详细介绍各种设计，对比各自的优劣，以便读者可以选择适合自己的方案。
- 第 10 章** 快速发布 各种团队的软件开发过程不尽相同，测试也各有差异。测试活动的开展依赖测试人员的多少、经验，软件的类型、规模、发布周期等多种因素。通过前面的章节，读者已经学习到如何让持续集成在团队中运转起来，但要让持续集成中的各个环节发挥出完全的功效，还需要将其中很多方面做细致。本章将分享在追求“完美”测试的道路上的亲身实践经验。本章前半部分，主要介绍从主线拉出分支到迭代发布的过程中，如何提高各个环节的质量和效率；后半部分，则重点讲述如何将测试延伸到一个迭代发布之后。希望对从事或者即将从事客户端测试的读者有所启发和帮助。

致 谢

非常感谢整个作者团队，牺牲了个人的休息时间，不断地修改文章直至满意为止。也感谢百度质量部出书小组对我们的支持，他们的出色工作使我们可以安静地写书。最后特别致谢出版社的符隆美女士，在我们写书过程中不断给予我们各种写书的经验，帮我们审稿。

目 录

上篇 技术篇

第 1 章 测试环境管理	1
1.1 问题的产生	1
1.1.1 客户端测试环境的概念	1
1.1.2 客户端测试环境管理需求	2
1.2 如何设计解决问题	3
1.2.1 环境如何自动化快照备份管理	3
1.2.2 应用软件如何自动化部署	7
1.2.3 环境资源池系统管理	9
1.3 实例展示	22
1.3.1 UI 功能展示	22
1.3.2 系统应用场景	24
1.3.3 项目应用效果	25
第 2 章 测试数据中心	27
2.1 为何要引入数据中心	27
2.1.1 概述	27
2.1.2 需求来源	28
2.1.3 功能分析	29

2.2	数据中心设计和实现	31
2.2.1	整体架构	31
2.2.2	数据存储设计和实现	32
2.2.3	测试辅助设计和实现	33
2.2.4	其他功能设计和实现	35
第 3 章	自动化测试	37
3.1	概述	38
3.1.1	什么是自动化测试	38
3.1.2	系统级自动化测试模型	38
3.2	基础库设计和实现	39
3.2.1	基础库设计原理	39
3.2.2	基于 Win 32 句柄	40
3.2.3	虚拟对象	45
3.2.4	基于 MSAA/UIA	52
3.2.5	后门&Proxy 模式	55
3.3	用例格式和开发模式	64
3.3.1	用例格式	65
3.3.2	基于脚本/代码的用例	65
3.3.3	开发模式	68
3.4	项目实例	71
3.4.1	基于虚拟控件的即时通信类产品系统级自动化	71
3.4.2	基于 Proxy 的播放器类产品系统级自动化	74
3.4.3	基于 Proxy 的浏览器类产品模块级 API 自动化	77
第 4 章	性能测试	80
4.1	性能指标	80
4.1.1	业务指标	81
4.1.2	系统指标	81
4.2	测试环境	85
4.3	测试方法及工具	86

4.3.1	业务指标	86
4.3.2	系统指标	92
4.4	测试场景与用例设计	94
4.4.1	普遍性	95
4.4.2	单一性	95
4.4.3	重复性	96
4.5	项目举例	96
4.5.1	场景与用例设计	96
4.5.2	测试执行	98
第 5 章	实网测试	103
5.1	为何要引入实网测试	103
5.1.1	实网测试概念	103
5.1.2	实网测试形式	104
5.1.3	实网测试关键因素	104
5.1.4	实网测试自动化系统解决方案	105
5.2	实网测试自动化系统设计实现	105
5.2.1	系统需求分析	105
5.2.2	系统整体设计	107
5.2.3	实网测试客户端	108
5.2.4	实网测试服务器	111
5.2.5	实网测试管理 Web	112
5.3	实例展示	113
5.3.1	实网测试系统实例	113
5.3.2	实网测试系统应用场景	116
5.3.3	实网测试收益分析	117
第 6 章	产品评测	120
6.1	评测的含义	120
6.1.1	评测概述	120
6.1.2	评测体系	124
6.1.3	评测作用	127

6.2	评测方案的产生	127
6.2.1	基本方法	127
6.2.2	输入法类产品实例	128
6.3	如何设计评测工具	135
6.3.1	评测工具需求分析	136
6.3.2	评测工具架构设计	137
6.3.3	评测数据生成接口	141
6.3.4	评测结果分析接口	141
6.3.5	评测工具/平台的处理流程	142

下篇 工程篇

第7章	敏捷及改变	145
7.1	什么是敏捷	145
7.2	为何走上敏捷之路	147
7.3	敏捷项目管理	148
7.3.1	敏捷项目管理之团队建设	148
7.3.2	敏捷项目管理之迭代规划	156
7.3.3	敏捷项目管理之需求管理	159
7.3.4	敏捷项目管理之计划跟踪	163
7.3.5	敏捷项目管理之信息传递	165
第8章	持续集成 (Continuous integration)	169
8.1	持续集成组成部分	169
8.1.1	持续集成简介	169
8.1.2	持续集成基本组成	170
8.1.3	配置管理	171
8.1.4	构建	173
8.1.5	测试	174
8.1.6	部署发布	174
8.2	持续集成的入门条件	174
8.3	支撑平台	176

8.4 Quick 构建	177
8.4.1 编译构建	177
8.4.2 Quick 构建目的	178
8.4.3 Quick 构建承载的测试类型	179
8.5 Slow 构建	183
8.6 Daily 构建	184
8.7 自动化部署	186
8.7.1 传统上线方式	186
8.7.2 自动化部署	188
第 9 章 构建过程优化	192
9.1 缩短测试执行时间	193
9.1.1 优化 Case	193
9.1.2 分布式执行	194
9.2 本地构建	199
9.2.1 为何需要本地构建	199
9.2.2 测试用例的选取	200
9.2.3 为何有了 Local Build，依然需要平台上的构建	200
9.2.4 本地构建实施方案	201
9.2.5 Local Build 实际应用情况	206
9.3 提高自动化用例执行的稳定性	208
9.3.1 引起自动化用例执行不稳定的因素	208
9.3.2 提高用例执行稳定性的方法	209
第 10 章 快速发布	211
10.1 如何去拉分支	212
10.1.1 冲突处理	213
10.1.2 经常同步	214
10.1.3 拉分支的标准	214
10.2 分支如何测试	216
10.2.1 系统级测试	216

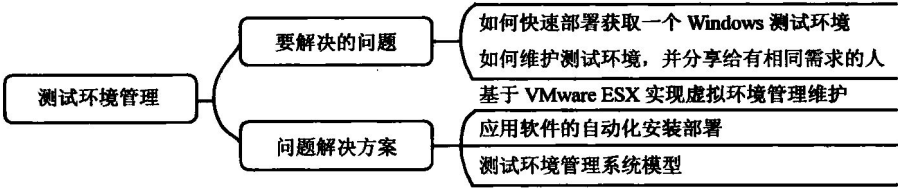
10.2.2	验收测试	218
10.3	发布的准备工作	219
10.3.1	逐级放量	219
10.3.2	放量指标	220
10.3.3	上线步骤测试	223
10.3.4	自动化上线	223
10.4	发布之后	225
10.4.1	下一个迭代	227
10.4.2	迭代后续	227

上篇 技术篇

第 1 章

测试环境管理

- 您是否遇到测试 Windows 软件时，要安装部署新操作系统或一堆兼容性软件的情况？如果有，花费多长时间？20 分钟、半个小时甚至更多？
- 可以在 10 分钟之内做到吗？
- 部署一个 Windows 测试环境之后，如何让其他人可以很轻松地复用呢？



1.1 问题的产生

1.1.1 客户端测试环境的概念

所谓客户端测试环境，是指部署在 Windows 操作系统下的应用软件测试环境，主要涉及操作系统和应用软件，测试环境应该能够满足被测客户端软件启动、运行、执行测试用例的基本操作，同时又能满足各种特殊测试用例的执行需求，如安全性测试用例、

兼容性测试用例等。

操作系统包括 Windows 主流系统版本，如 Windows XP、Windows Server 2003、Windows Vista、Windows 7、Windows Server 2008，版本包括中文和英文等。

应用软件按类别大概分为即时通信、输入法、浏览器、网络下载、视频播放、图片工具、安全防护、系统工具、办公学习、开发工具、股票网银、游戏休闲等。

当测试客户端软件时，需要根据软件产品设计、需求设计编写测试用例。在用例设计过程中，需要考虑用例的执行测试环境，比如用例 A 需要在 Windows XP 系统下执行，用例 B 需要在安装诺顿杀毒软件的环境下执行，用例 C 执行环境为 Windows 7+XX 网游+卡巴斯基等。

我们的测试用例执行环境越丰富、越接近大部分主流用户环境，才能够更大程度地保证客户端软件满足用户的正常使用。

1.1.2 客户端测试环境管理需求

客户端测试环境管理的最主要需求是提高测试效率，测试环境的高效管理对于提升工作效率至关重要。

当执行测试用例需要某个特殊的测试环境时，一般情况下简单测试环境手工部署平均时间在 10 分钟~30 分钟，部分复杂环境部署约数小时，比如需要安装部署某个版本的操作系统或下载安装某个大型杀毒软件等。此时环境部署的时间往往超过测试用例执行的时间，最让我们头痛的不是测试本身，而是测试环境的搭建。

当执行完毕某个测试用例时，往往会因为执行其他用例的需要而改变已搭建好的测试环境，这样下次回归需要重新手工部署搭建，这些重复劳动极大降低了测试效率。如果能够自动保存我们辛苦部署的测试环境，下次可以直接复用就好了。

Windows 系统下的应用软件数量和种类不胜枚举，初次搭建环境时手工下载安装这些应用软件也是极为耗时的一项工作，想一想我们需要到各大软件下载站，找到各个版本的应用软件，然后辛苦下载下来，再逐个安装。多么枯燥乏味的工作！如果能够将这部分工作全部自动化实现，会极大节省部署时间。

另外，在一个团队人员众多，而硬件测试机器资源有限的情况下，想想大家争抢测试机器的局面吧！我们迫切需要一个管理系统，来帮助大家协调测试机器的占用问题。比如先占用了环境的人，有权使用固定时间，这段时间别人不能强占；到期后别人可以抢占，大家有序竞争，至少不会发生环境使用的互踢现象。

综上，客户端测试环境的管理需求如下。