

Web自动化测试的专业工具书，
软件测试工程师的良师益友。

零成本实现 Web自动化测试

——基于Selenium和Bromine

温素剑 编著



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

零成本实现 Web自动化测试

——基于Selenium和Bromine

温素剑 编著

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书从目前主流的 Web 自动化测试工具入手,由浅入深、全面、细致地阐述了如何使用开源测试工具来完成 Web 自动化测试,便于读者轻松掌握 Web 自动化测试的原理、方法和实际操作。所以,它不是一本深奥的测试理论书籍,而是一本测试实战类的书籍,作者试图在书中向读者演示如何使用 Selenium、WebDriver、Bromine、SauceLabs、MantisBT 搭建起一整套稳定、高效、成本为零的自动化测试平台。本书以测试 Google GMAIL 为具体范例,提供了完整的 Web 自动化测试解决方案。

本书是一本 Web 自动化测试的专业工具书,也是软件测试工程师的良师益友,可作为软件测试培训、高等院校软件相关专业的辅助教材,目标读者是初级或者资深软件测试工程师、对提高自动化测试的有效性感到困惑的测试经理,以及有志于从事软件测试工作的应届毕业生。

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

零成本实现 Web 自动化测试:基于 Selenium 和 Bromine / 温素剑编著. —北京:电子工业出版社, 2011.5

ISBN 978-7-121-13205-6

I. ①零… II. ①温… III. ①软件工具—测试 IV. ①TP311.56

中国版本图书馆 CIP 数据核字(2011)第 053062 号

策划编辑:郭立 袁金敏

责任编辑:郭立

特约编辑:顾慧芳

印 刷:

装 订:北京中新伟业印刷有限公司

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:24.5 字数:450 千字

印 次:2011 年 5 月第 1 次印刷

印 数:4 000 册 定价:59.00 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 zltts@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

我有一个梦想，有一天我的办公桌上不再是一片狼藉，整洁的台面上只有一个大大的红色按钮，每天最重要的工作，就是上班后狠狠地按下红色按钮，然后悠闲地泡上一杯西湖龙井，拿起最新一期的参考消息，看看美联储是不是又在加印该死的“废纸”。在我享受生活的同时，一份份准确严谨的测试报告就会自动发往开发部门，并抄送到各位领导的邮箱中。测试报告包含最新软件版本的缺陷列表，并指出了可接受的缺陷修复时间。剩下的一切都是“傲慢自大”的软件开发人员的事，多么美妙的一天。

不过梦想永远都是梦想，就像马丁·路德·金遇刺身亡 42 年后的今天，美国仍然不是一个人人平等、没有种族歧视的国家一样。如果我的梦想有一天变为现实，那么我需要面对的可能不是悠闲的生活，而是一份解聘通知书。因为真到那一天，软件测试工作就变成人人都能完成的简单事情，公司完全可以随便请一个人来负责我的工作，而不必付出很高的薪水。这真是一个一点儿也不好笑的笑话，测试人员的终极目标居然是让自己失去饭碗，不过，幸好这一天还远着呢。

现在，让我们从梦想世界回到现实中来，现实是残酷的，笔者所在的测试部门，已经拥有成千上万个 QTP 自动化测试案例，不过它们仅仅是自动化测试案例而已。梦想离笔者还远着呢，这些案例总是执行失败，失败原因千奇百怪，既有测试环境不稳定导致的失败，也有 QTP 工具莫名其妙的错误。测试工程师投入了大量的精力去维护这些自动化测试案例，可惜效果并不好，自动化测试的有效率依然很低，很多软件系统的自动化测试有效率常年保持在 60% 左右，怎么办？如何向领导解释？

要提高自动化测试的有效性，笔者认为有两种办法：

第一，降低自动化测试对象的层次，即采用底层接口自动化测试。一个软件系统的上层界面可能会经常发生变化，但是底层功能模块之间接口和系统之间接口的变化却一定不会很多，除非遇到软件重构。这样做就能在很大程度上避免了用户界面频繁变化对自动化测试的影响，从而降低了维护的工作量。

第二，采取更可靠的基于 GUI 的自动化测试工具。目前 QTP 无疑是使用最广泛的自动化测试工具，但是实践中笔者发现它的录制/回放功能很不稳定，经常会发生测试对象无法识别的异常情况，甚至莫名其妙地运行失败。怎么办？放弃基于用户界面的自动化测试？显然不行，因为底层接口的自动化测试，有其自身的局限性，首先你无法通过它来测试用户界面，其次你很难用它来测试软件系统的完整业务流程。唯一可行的办法就是寻找一种更可靠、更有效的工具来代替它，在这里笔者推荐 Selenium 和 WebDriver 两种开源的自动化测试工具。从目前的实践来看，它们远比 QTP 更稳定，测试案例的执行速度也更快，使用成本会更低。

写作背景

最近几年自动化测试已日益深入人心，如果现在有人去面试软件测试工程师岗位，而又完全不了解自动化测试，那么他几乎就不可能得到职位。因此，对测试工程师而言，了解自动化测试原理、掌握几种主流的自动化测试软件，已经变成自动化测试岗位最基本的要求。对有志于从事软件测试行业的大学应届毕业生而言，了解自动化测试就是一块不错的敲门砖。今年笔者所在的测试部门来了两位实习生，由笔者辅导他们完成一些自动化测试任务。实习结束后他们并没有与公司签约，后来与他们沟通才知道，别的公司肯出双倍的薪水聘用他们，看重的正是他们有自动化测试的工作经验。由此，读者朋友们就不难看出业界对自动化测试的重视程度。

“云计算”绝对是 2010 年最热的 IT 词汇，甚至沾上一点“云”概念的股票都会一飞冲天。“云”听起来很虚幻，其实就是瘦客户端加网格计算。今后客户端不再会有大量的计算任务，计算和存储都被放在云上。在笔者看来，今后的客户端应该就

零成本实现 Web 自动化测试——基于 Selenium 和 Bromine

是一个浏览器，用户所有的操作都是通过浏览器来实现的。Google 刚发布的操作系统 Chrome OS，就是基于这一理念设计的。B/S 和 C/S 架构的软件系统，应该会慢慢演变为 Browser/Cloud 模式。如此看来在“云计算”时代，Web 自动化测试依然很重要，而且会越来越重要。因此，笔者萌生了写作一本关于 Web 自动化测试的书籍。

本书内容

本书不是一本讲述深奥测试理论的教科书，而是一本实战性的书籍。笔者想要达到的目标是——读者朋友们在认真读完这本书后，马上就能在生产实践中用上所学的内容。本书首先介绍基础的自动化测试理论，接着介绍 Selenium 和 WebDriver 两种测试工具，最重要的是自动化测试实战篇。实战篇中笔者以测试 Google Gmail 为例，手把手地教读者如何用 Selenium、WebDriver、Bromine、SauceLabs、MantisBT 搭建起一整套稳定、高效、成本为零的自动化测试平台。

本书分为三篇共 14 章。第 1 篇包括第 1~9 章，第 1 章介绍自动化测试理论，第 2 章到第 8 章介绍 Selenium 自动化测试工具，第 9 章介绍 WebDriver 自动化测试工具。第 2 篇是自动化测试实战篇，包括第 10~13 章，第 10 章以测试 Google GMAIL 为范例介绍完整的 Web 自动化测试解决方案，第 11 章介绍管理测试案例的工具——Bromine，第 12 章介绍测试云 SauceLabs，第 13 章介绍基于 Web 的缺陷管理系统——MantisBT。第 3 篇是 Selenium 使用帮助篇，是使用 Selenium 经常需要用到的一些资料。

目标读者

本书的目标读者是初级或者资深软件测试工程师，以及对提高自动化测试的有效性感到困惑的测试经理。本书着重介绍如何使用 Selenium 和 WebDriver 两种开源自动化测试工具，来构建你的 Web 自动化测试体系。

感谢

首先要感谢我的家人，正是由于他们默默的支持，我才能静下心来写作。其次，还要感谢电子工业出版社计算机分社社长郭立和策划编辑袁金敏的热情支持和帮助，以及其他编辑认真细致的工作才能使书稿得以完善和出版。

由于笔者水平有限，书中的错误和不要之处在所难免，敬请读者批评指正。有关本书的反馈和咨询，可以发送邮件至 wensujian815@gmail.com。

温素剑

2010 年 12 月 16 日于北京

目录

第 1 篇 Selenium 和 WebDriver 工具篇

第 1 章 自动化测试基础	2
1.1 初识自动化测试	2
1.1.1 自动化测试是什么	2
1.1.2 常见的自动化测试架构	5
1.1.3 自动化测试的基本流程	8
1.1.4 走出自动化测试认知的误区	12
1.2 开源 Web 自动化测试	14
1.2.1 Web 自动化测试	14
1.2.2 是否选择开源自动化测试	15
1.2.3 开源自动化测试的优势	15
第 2 章 Selenium 简介和基础	17
2.1 Selenium 简介	17
2.2 Selenium 基础	21
2.3 小结	25
第 3 章 Selenium-IDE	26
3.1 简介	26
3.2 IDE 操作界面	29

3.3 测试案例和测试案例集	32
3.3.1 创建测试案例	32
3.3.2 运行测试案例	35
3.3.3 调试测试案例	37
3.3.4 测试案例集	39
3.4 IDE 功能扩展	40
3.5 常见错误处理	41
第 4 章 Selenium 命令	45
4.1 验证页面元素	45
4.2 定位页面元素	48
4.3 文字范本匹配	54
4.4 命令使用技巧	58
第 5 章 Selenium-RC	63
5.1 简介	63
5.2 Selenium-RC 基础	64
5.2.1 Selenium-RC 工作原理	64
5.2.2 Selenium-RC 架构	66
5.2.3 Selenium-RC 安装	70
5.3 Selenium-RC 使用	74
5.3.1 从 Selenese 到测试程序	74
5.3.2 构建你的测试程序	79
5.3.3 学习 Selenium-RC API	86
5.3.4 报告测试结果	88
5.3.5 为测试加入一些编程逻辑	91
5.3.6 Server 选项	94
5.3.7 处理 HTTPS 和安全弹出对话框	99
5.3.8 支持其他浏览器和浏览器配置	100
5.4 常见错误的处理	101

第 6 章 Selenium 测试案例设计	107
6.1 测试设计导言	107
6.2 测试什么	107
6.3 验证结果	109
6.3.1 断言 vs 验证	109
6.3.2 权衡: assertTextPresent、assertElementPresent、assertText	110
6.4 选择一个定位策略	111
6.4.1 定位动态对象	111
6.4.2 如何避免在测试中使用复杂的 xpath 表达式	114
6.5 测试 Ajax 应用系统	115
6.6 UI 映射	115
6.7 数据驱动测试	118
6.8 数据库验证	119
第 7 章 Selenium 用户扩展	121
7.1 简介	121
7.2 Actions	121
7.3 Accessors/Assertions	122
7.4 定位策略	123
7.5 在 Selenium-IDE 中使用用户扩展	124
7.6 在 Selenium-RC 中使用用户扩展	124
第 8 章 Selenium-Grid	126
8.1 Selenium-Grid 是如何工作的	127
8.1.1 传统的 Selenium 架构	127
8.1.2 Selenium-Grid 架构	128
8.1.3 指定测试执行环境	130
8.2 如何使用 Selenium-Grid 测试平台	136
第 9 章 Selenium 2.0 和 WebDriver	142
9.1 WebDriver 基础	142

9.1.1	WebDriver 是什么	142
9.1.2	什么时候使用 WebDriver	143
9.1.3	五分钟入门指南	143
9.2	WebDriver 的各种实现	146
9.2.1	HtmlUnit Driver	146
9.2.2	Firefox Driver	147
9.2.3	Internet Explorer Driver	149
9.2.4	Chrome Driver	149
9.3	WebDriver 进阶	151
9.4	模拟 Selenium-RC	156

第 2 篇 自动化测试实战篇

第 10 章 实战——测试 Google GMAIL

160

10.1	明确测试范围	160
10.2	Selenium 测试——登录 GMAIL 邮箱	161
10.3	Selenium 测试——撰写邮件	172
10.4	Selenium 测试——删除邮件以及邮件列表	178
10.5	Selenium 测试——邮件存档	185
10.6	Selenium 测试——搜索邮件	191
10.7	使用测试案例集来管理 Selenium-IDE 脚本	197
10.8	留给读者的思考	200

第 11 章 管理你的测试案例——Bromine

201

11.1	Bromine 基础	201
11.1.1	为什么使用 Bromine	201
11.1.2	Bromine 特性	202
11.1.3	Bromine 工作原理	202
11.1.4	安装 Bromine	204
11.2	配置和使用 Bromine	206

零成本实现 Web 自动化测试——基于 Selenium 和 Bromine

11.2.1	配置 Bromine 测试节点.....	206
11.2.2	配置 Bromine 任务调度器.....	209
11.2.3	创建一个 Bromine 测试脚本.....	212
11.2.4	在 Bromine 中上载测试脚本.....	216
11.2.5	测试计划.....	216
11.2.6	测试实验室.....	219
11.2.7	控制面板.....	220
11.3	实战——管理 Google GMAIL 测试案例	222
第 12 章 测试云——SauceLabs		229
12.1	Sauce OnDemand.....	229
12.1.1	如何使用 Sauce OnDemand.....	231
12.1.2	Sauce OnDemand 常用参数.....	238
12.2	Sauce Connect.....	246
12.3	Sauce IDE.....	248
12.4	Sauce RC	253
12.5	实战——在 SauceLabs 中运行 Google GMAIL 测试案例	258
12.6	实战——在 Bromine 中集成 SauceLabs	267
第 13 章 管理你的缺陷——MantisBT		271
13.1	简介	271
13.2	安装	273
13.3	MantisBT 配置.....	276
13.4	MantisBT 管理.....	280
13.4.1	用户管理.....	280
13.4.2	项目管理.....	283
13.4.3	自定义字段管理.....	285
13.4.4	管理配置.....	286
13.5	报告缺陷	289
13.5.1	报告并处理缺陷.....	290

13.5.2	缺陷查询.....	291
13.5.3	缺陷修改.....	292
13.5.4	我的视图.....	294
13.6	统计报表和图形报表.....	294
13.7	留给读者的思考.....	297

第 3 篇 Selenium 使用帮助篇

第 14 章 Selenium 使用帮助 300

14.1	Selenium API 速查手册.....	300
14.1.1	Selenium Action.....	300
14.1.2	Selenium Assertion	318
14.1.3	Selenium Accessors.....	320
14.2	Selenium 客户端驱动配置.....	353
14.2.1	.NET 客户端驱动配置.....	353
14.2.2	Java 客户端驱动配置.....	356
14.2.3	Python 客户端驱动配置	372

第 1 篇

Selenium 和 WebDriver 工具篇

- ▶ 第 1 章 自动化测试基础
- ▶ 第 2 章 Selenium 简介和基础
- ▶ 第 3 章 Selenium-IDE
- ▶ 第 4 章 Selenium 命令
- ▶ 第 5 章 Selenium-RC
- ▶ 第 6 章 Selenium 测试案例设计
- ▶ 第 7 章 Selenium 用户扩展
- ▶ 第 8 章 Selenium-Grid
- ▶ 第 9 章 Selenium2.0 和 WebDriver

第 1 章

自动化测试基础

1.1 初识自动化测试

1.1.1 自动化测试是什么



维基百科对自动化测试的定义——自动化测试就是使用软件来控制测试案例的执行。它将实际测试结果与预期结果进行比较，并提供测试预置条件设定、测试逻辑控制以及测试报告等重要功能。通常，自动化测试都是基于已经存在，并拥有成熟测试流程的人工测试来实现的。

尽管人工测试可以发现大量的软件系统缺陷，但它无疑会耗费大量人力及时间，并且不容易发现某些特定类型的缺陷（例如数个下拉菜单的逻辑组合查询测试）。对测试进行自动化的过程，就是写一段程序来代替人工测试。一旦测试被成功自动化后，它们就可以在任何需要的时候快速执行。对于软件维护周期很长的应用系统而言，这

零成本实现 Web 自动化测试——基于 Selenium 和 Bromine

无疑是最高效的方式。因为在漫长的软件维护周期中，一个很小的改动，都可能导致原本正常的软件功能失效。

自动化测试有两种常见的方式：

1. 代码驱动测试，通过大量不同的输入参数和对应的返回结果，来验证类、模块或者库文件的公共接口是否正确。目前软件开发行业有一个流行趋势，那就是普遍使用测试框架（例如 xUnit 框架、JUnit 和 NUnit）来进行单元测试，以便于判断代码片段在不同配置环境下的表现是否符合预期。测试案例被测试人员用来描述如何操作待测程序，以便验证程序表现是否符合预期。

“代码驱动测试自动化”是敏捷开发的一个关键特性，同时它又被很多人称为测试驱动开发（TDD）。单元测试的逻辑在代码编写之前就应该确定下来，只有当代码通过了所有测试，代码编写工作才能结束。支持这一看法的业界专家认为，以“代码驱动测试自动化”方式测试通过的代码，比传统人工测试通过的代码更可靠，代码的维护成本也会更低。原因在于“代码驱动测试自动化”的代码覆盖率更高，另外在整个软件开发周期中它都可以不断地进行测试，而不必等到传统瀑布开发模型的最后阶段。开发人员可以在每次代码变动之后，立刻发现缺陷，而不必等到测试阶段去修复，这样成本会低得多。

2. 图形用户接口测试，测试框架产生用户接口事件（例如键盘敲击、鼠标单击等），并捕获事件导致的图形用户接口改变，以便验证可见的程序响应是否正确。很多自动化测试工具都支持录制/回放特性，它们允许用户交互性地录制自己的操作，在需要的时候进行回放，并将回放的实际结果与期望结果进行比较。这一方式的优点在于，它不需要大量的开发工作，甚至根本没有开发工作量。它可以被用于测试任何拥有图形界面的软件系统，但是如果你依赖于这种方式，就会面临测试可靠性存疑和测试脚本不易维护的问题。有时候一个按钮在窗口中移动了位置，都可能导致需要重构测试脚本。录制/回放特性有时候还会在测试脚本中加入不相关的用户操作，甚至不能正确录制某些操作。这类自动化测试的一个变种，就是 Web 自动化测试，它的测试对象是 Web 页面，而非 Windows 程序界面。它读取 HTML 文件，而非 Windows 事件，例如，Selenium 和 WebDriver。这类自动化测试还有另一个变种，那就是无脚本自动化测试，它并不使用录制和回放，而是去建立待测系统的模型。在这种模式下，

测试人员建立测试案例，只需提供测试参数和条件。测试人员无须编写测试脚本，就拥有了测试脚本的基本功能和灵活性。测试案例维护比较简单，因为没有脚本需要你去维护，待测系统的对象发生改变后，可以很容易地重新识别或者添加。它适用于任何基于 GUI 的软件系统。例如 QTP 的关键字驱动。

本书只包含图形用户接口测试的内容，对代码驱动测试感兴趣的读者，可以查阅相关书籍，或者在互联网上搜索。

自动化测试工具通常都是很昂贵的，并经常与手工测试结合起来使用。如果在较长的时间周期内考察，那么自动化测试是有价值的，它特别适合于定期的回归测试。

在开始自动化测试工作前，您必须先决定自动化什么，什么时间自动化，甚至于是是否需要自动化。选择待测软件系统的哪些功能特性进行自动化测试，在很大程度上决定了自动化工作的成败。对不稳定的功能特性，或者即将改变的功能特性，进行自动化测试，必定得不偿失，绝对不要这么做。

是否需要测试进行自动化？

自动化总是有利可图的吗？什么时候我们会决定对测试案例进行自动化？

自动化并非总是有利可图的，有时候人工测试更适合。举个例子，如果系统的用户界面会在最近发生改变，那么自动化测试案例就不得不面临重构。而有时候并没有足够的时间去设计自动化测试，对于工期较短的项目，人工测试也许更有效率。如果一个项目有严格的截至时间点，并且当前没有可用的自动化测试案例，那么人工测试是你最好的解决方案。

不过自动化测试在一个长周期的迭代测试中，拥有许多优点：

- 能够支持频繁的回归测试；
- 能够在软件开发过程中尽早发现缺陷；
- 能够无限地循环执行测试案例；

低成本实现 Web 自动化测试——基于 Selenium 和 Bromine