

博为峰旗下

51testing

软件测试网 作品系列

软件功能测试 自动化实战教程

Practices in Advanced Automated Functional Software Testing

51Testing软件测试网 组编

周焕来 贾海涛 编著

- 全面讲解软件功能测试理论、技术，以及工具的用法
- 引入安全测试中的Fuzzing测试思想，形成完整、便捷的软件功能测试解决方案
- 基于最新的软件功能测试工具UFT 11.5讲解，包括QTP脚本开发等
- 引入了Peach等功能测试工具，建立了一套完整的、跨平台的功能测评体系

 人民邮电出版社
POSTS & TELECOM PRESS

博为峰旗下

51testing

软件测试网 作品系列

软件功能测试 自动化实战教程

Practices in Advanced Automated
Functional Software Testing

51Testing软件测试网 组编

周焕来 贾海涛 编著

人民邮电出版社

北京

图书在版编目 (C I P) 数据

软件功能测试自动化实战教程 / 周换来, 贾海涛编
著; 51Testing软件测试网组编. — 北京: 人民邮电出版社, 2015.3

ISBN 978-7-115-38471-3

I. ①软… II. ①周… ②贾… ③5… III. ①软件—
测试—自动化—教材 IV. ①TP311.5

中国版本图书馆CIP数据核字(2015)第023956号

内 容 提 要

软件功能测试既是基础的软件测试领域, 也是一个没有完全解决问题的领域, 若要便捷有效地进行自动化功能测试是一件不简单的事情, 为了帮助读者较好地地进行自动化功能测试, 特意撰写本书, 本书的主要内容为: 第一部分理念篇, 主要包括软件测试概论、软件自动化测试、数据驱动测试技术、数据驱动测试技术的特点、数据驱动的自动化测试框架、数据驱动测试中的参数化技术、数据驱动测试的应用、关键字测试技术、关键字驱动测试特点、关键字驱动的自动化测试系统实现等; 第二部分 GUI 测试篇, 包括 Unified Functional Testing 介绍、分析应用程序、调试和分析测试、使用 GUI 测试、编写自动化测试脚本、GUI 的自动化测试方法、在关键字视图中添加测试信息的输出、编辑测试脚本、运行和调试测试脚本、利用活动屏幕来编辑测试脚本、关键字驱动测试方法、关键字驱动测试步骤; 第三部分 API 与服务测试篇, 包括 API/接口测试、UFT 中的 API 测试、API 测试通用流程、Web Service 测试; 第四部分非主流测试工具篇, 主要有 Linux 系统交互应用功能测试框架工具 Expect、网络协议自动化测试工具 Sulley、创建 FTP 协议描述框架、无界面应用功能自动化测试工具 Peach 等技术。

本书适合软件测试从业者、新入职人员, 以及大专院校相关专业的师生用书和培训学校的教材。

◆ 组 编 51Testing 软件测试网

编 著 周换来 贾海涛

责任编辑 张 涛

责任印制 张佳莹 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>

三河市潮河印业有限公司印刷

◆ 开本: 800×1000 1/16

印张: 16.25

字数: 293 千字

印数: 1-3 000 册

2015 年 3 月第 1 版

2015 年 3 月河北第 1 次印刷



定价: 45.00 元

读者服务热线: (010)81055410 印装质量热线: (010)81055316

反盗版热线: (010)81055315

编辑委员会名单

编委

- 王威（具有多年软件开发经验和软件测试工作经验，对产品测试的方法和管理流程有深刻的认识，51Testing 软件测试培训高级讲师）
- 王琰（具有丰富的通信终端产品的测试以及管理工作经验，51Testing 软件测试培训高级讲师）
- 朴春龙（Mercury 认证 CPC，自动化测试专家，51Testing 软件测试培训高级讲师）
- 邓强（具有丰富的软件系统测试工作经验，对功能性测试、自动化测试、性能测试及测试框架均有深入的研究，51Testing 软件测试培训高级讲师）
- 周峰（信息产业部认证系统分析员，51Testing 软件测试培训高级讲师）
- 周春江（具有多年通信协议和通信终端设备的测试工作经验，51Testing 软件测试培训高级讲师）
- 商莉（多年从事软件开发、软件测试及质量保证方面的管理工作，51Testing 软件测试培训高级讲师）
- 宋光照（具有深厚的开源软件测试实践经验，擅长嵌入式测试，51Testing 软件测试培训高级讲师）
- 刘德宝（具有多年软件测试及培训经验，参与门户网站、移动 BOSS 等多个大型软件测试项目，在自动化测试技术方面具有丰富的项目经验，51Testing 软件测试培训高级讲师）

编辑部成员

张晓晓 严代丽

自序

——软件测试迷局

我是一名在软件研发与测试产业中工作超过 15 年的从业者，在一个偶然的时机中从软件研发领域转入软件测试领域，并承担起了软件测试领域的教学和研究工作，常常感叹人生不是能计划出细节的，生活充满了迷一样的色彩与神奇。但这样的感悟似乎用在软件测试产业中也是很恰当的，软件测试产业也拥有着同样的迷一样的色彩！

软件测试产业的诞生时间就是个迷，目前没有人能说清楚软件测试产业具体是在哪一天诞生的。总的说来软件测试产业是一个既古老又年轻的行业，说它古老，是说它跟软件研发产业一起诞生，20 世纪 50 年代就有了；但是又可以说它无比年轻，直到 20 世纪 80 年代末软件工程学科确立（软件研发才正式有了生命周期模型与阶段），软件测试才作为一个软件工程过程有了独立的价值与内涵，成为了一个独立领域。随后软件研发产业得到了快速发展，日益成为国民经济中不可或缺的要害，软件测试也日益发展成一个独立的产业，到今天已经逐渐发展成为了几乎与软件研发并驾齐驱的一个产业领域了，没有人怀疑缺乏了测试环节的软件研发是残废品，风险多多。所以，软测产业的诞生与发展到底只有一二十年的历史，还是覆盖了整个软件产业发展的历史，这是个说不清楚的迷。

软件测试的迷样色彩远不止诞生历史这样一点，软件测试是一个同时被尊重和贬低的领域，充满了矛盾。从软件测试领域的重要性来说，但凡是严肃意义上的软件产业从业者，大家对软件测试的重要意义是百分百地认可和尊重的，说到软件测试的时候不乏溢美之词；但是任何一个严肃意义上的软件从业者也不能否定目前软件测试领域的基本态势是做不了软件研发的人员做软件测试，这个产业的实践实际上是在贬低软件测试的重要性。这个矛盾以及为什么形成了这样一个矛盾局面，不是三言两语可以说清楚的，软测产业的混沌与迷在我进入软件测试领域的这些年也还没有看到明显的改变迹象。

再谈一谈软件测试的实践。近些年来软件测试领域涌现了大量的技术和工具，从测试核心支撑技术到辅助监控分析技术，不可谓不全面，软件测试的实践已经变得简单轻松起来，但是同时我们也发现，软件测试在最基本的测试实践上也还存在不少问题，比如在功能自动化测试领域——这个在软件测试产业里公认的最成熟的领域——依然还存

在重大问题，有很多软件全部或部分的功能测试无法进行自动化测试，进而造成了测试必须后退到手工的完全原始的模式进行（这也是我编写本书的原由）。软件测试产业的技术进步与待解决的问题总是同样多，这样的局面，迷惑了软件产业的多数从业者，大家时而对软件测试充满了信心，觉得软件测试可以解决很多问题；时而对软件测试充满了困惑，觉得为什么在如此基础的问题上，软件测试都还如此不成熟而不为所用。软件测试产业在核心的实践环节上也具有着迷一样的色彩。

作为一名资深的软件研发人员和一个软件测试从业人员，我对以上几点迷局的看法是这样的：第一，还没有消除人为认知上的一些对软件测试产业相对肤浅的歧视，这主要来源于对软件测试中简单的手工测试的印象；第二，软件测试技术的发展还没有成熟完备到可以解决由软件研发所产生的全部问题，很不幸的是可能软件测试从根本上就有些问题不能突破，毕竟软件测试是逆向过程，逆向过程一般不能还原到正向的全部过程。

在这段文字中我使用了“迷局”这样的字眼，倾向似乎悲观，但我确实不是一个悲观主义者，我认为软件测试产业确实存在着这样的迷局或者说矛盾局面，作为从业者我们否定和美化这样的局面是没有意义的，只有通过从业人员不断地发挥专业精神获得行业和产业的认可与尊敬，这些迷局才能真正清澈起来，这是我们软件测试领域的进步动力也是我们大家作为软件测试从业者义不容辞的责任！

周焕来

前言

近些年来软件测试领域涌现了大量的技术和工具，从测试核心支撑技术到辅助监控分析技术，这使软件测试的实践已经变得简单轻松起来，但是，我们也发现，软件测试在最基本的测试实践上也还存在不少问题，比如在功能自动化测试领域——这个在软件测试产业里公认的最成熟的领域——依然还存在重大问题，有很多软件全部或部分的功能测试无法进行自动化测试，进而造成了测试必须后退到手工的、完全原始的模式进行（这也是我编写本书的初衷）。软件测试产业的技术进步与待解决的问题总是同样多，这样的局面，迷惑了软件产业的多数从业者，大家时而对软件测试充满了信心，觉得软件测试可以解决很多问题；时而对软件测试充满了困惑，觉得为什么在如此基础的问题上，软件测试都还如此不成熟而不为所用。

目前，同类书籍主要围绕 QTP 的使用展开，本书主要围绕 QTP 脚本开发展开，不局限在 QTP 工具本身。本书用到了 HP QTP 工具的最新版本 UFT 11.5 的资料，另外，立足于完整的功能测试，引入了 Peach 等非常见功能测试工具资料，建立了一套完整的、跨平台的功能测评体系。

本书的主要内容为：首先介绍软件功能测试中的两大关键理念：数据驱动测试和关键字测试；然后围绕这两大理念全面介绍了经典测试工具 UFT11.5（QTP 的最新版本）的功能和使用方法，并结合实践讲解具体的使用方法及实现数据驱动测试和关键字测试的方法；最后，在 UFT 的基础上引入几个高阶工具，并讲解了安全测评中的 Fuzzing 测试，并引入到功能自动化测试领域，补充常用功能测试工具在非界面测试和非 Windows 平台测试方面的不足。

本书适合软件测试从业者以及大专院校相关专业的师生使用，也可以作为相关培训学校的教材。

编辑联系邮箱：zhangtao@ptpress.com.cn。

目 录

第一部分 理念篇

第 1 章 软件测试概论	2
1.1 软件测试概论	2
1.1.1 软件测试概述	2
1.1.2 国内外现状	3
1.1.3 软件测试与开发的关系	5
1.1.4 软件测试基础	7
1.2 软件自动化测试	17
1.2.1 软件自动化测试概述	18
1.2.2 软件自动化测试的优势	18
1.2.3 软件自动化测试的局限	19
1.2.4 自动化测试的分类	20
1.3 自动化测试的发展趋势与境界	23
第 2 章 数据驱动测试技术	25
2.1 数据驱动测试技术概述	25
2.1.1 数据驱动测试的概念	25
2.1.2 数据驱动测试技术的来源	25
2.1.3 数据驱动测试技术的特点	26
2.1.4 数据驱动测试技术的适用测试阶段	26
2.1.5 数据驱动测试工具支持情况	27
2.2 数据驱动测试的关键内涵	28
2.2.1 数据驱动的自动化测试框架	28
2.2.2 数据驱动测试的流程	30
2.2.3 数据驱动测试中的驱动脚本	30
2.2.4 数据驱动测试中数据的选择	30
2.2.5 数据驱动测试中的参数化技术	32

2.2.6	数据驱动测试中的数据源	34
2.3	数据驱动测试的应用	35
2.3.1	同步数据驱动测试 (CSDDT)	35
2.3.2	数据驱动引擎	36
2.4	数据驱动的优点和不足	36
第 3 章	关键字测试技术	38
3.1	关键字驱动测试概述	38
3.1.1	概念	38
3.1.2	关键字驱动测试来源	38
3.1.3	关键字驱动测试特点	39
3.1.4	适用的测试阶段	40
3.1.5	主要工具支持情况	40
3.2	关键内涵	41
3.2.1	关键字设计	41
3.2.2	关键字驱动的自动化测试系统实现	43
3.2.3	方法	46
3.3	技术优点和不足	46

第二部分 GUI 测试篇

第 4 章	Unified Functional Testing 介绍	49
4.1	简介	49
4.2	测试的基本过程	49
4.2.1	分析应用程序	49
4.2.2	准备测试框架	50
4.2.3	构建测试和添加每个测试步骤	50
4.2.4	强化测试过程	50
4.2.5	调试、运行和分析测试	50
4.2.6	报告缺陷	51
4.3	开始使用 GUI 测试	51
4.3.1	启动	51
4.3.2	插件加载设置与管理	51
4.3.3	创建一个测试项目	52

4.3.4	录制和测试运行设置	53
4.3.5	指定需要录制的应用程序	54
4.3.6	编写第一个自动化测试脚本	56
第 5 章	GUI 的自动化测试方法	58
5.1	GUI 的使用	59
5.1.1	自动化测试的基本过程	59
5.1.2	在关键字视图中编辑测试脚本	59
5.1.3	在关键字视图中为测试步骤添加注释	60
5.1.4	在关键字视图中添加测试信息的输出	60
5.1.5	插入检查点 (Checkpoint)	61
5.1.6	在关键字视图中插入新的测试步骤	62
5.1.7	在专家视图中编辑测试脚本	65
5.1.8	脚本编辑器的使用	65
5.2	编辑测试脚本	66
5.2.1	识别对象	66
5.2.2	Object Spy 的使用	67
5.2.3	对象库管理	69
5.2.4	把测试对象添加到对象库中	69
5.2.5	导出对象库文件	71
5.2.6	在测试脚本中访问对象库的测试对象	71
5.2.7	新的 Action	73
5.2.8	关联 Action 的对象库	74
5.3	运行和调试测试脚本	75
5.3.1	语法检查	75
5.3.2	使用断点	76
5.3.3	单步调试	77
5.3.4	运行整个测试	77
5.4	分析测试结果	78
5.4.1	选择测试运行结果的存储位置	78
5.4.2	查看概要测试运行结果	79
5.4.3	查看检查点的结果	79
5.4.4	查看测试过程的截屏	80
5.5	利用活动屏幕来编辑测试脚本	81
5.5.1	查看活动屏幕	81

5.5.2	活动屏幕的设置	81
5.5.3	在活动屏幕中选择并添加对象到对象库	82
5.5.4	选择活动屏幕中的对象并输出其对象属性	84
5.5.5	在活动屏幕中插入检查点	86
5.5.6	在活动屏幕中选择对象并添加测试步骤	89
第 6 章	数据驱动测试	91
6.1	什么时候使用数据驱动测试方法	91
6.2	数据驱动测试的一般步骤	91
6.3	参数化测试	92
6.3.1	通过参数化测试来提高测试的灵活性	92
6.3.2	参数化测试步骤	93
6.3.3	使用随机数来进行参数化	95
6.3.4	参数化检查点	97
6.3.5	设置数据表格迭代方式	98
6.4	Action 测试输入的参数化	99
6.4.1	编辑 Action 的属性	99
6.4.2	添加 Action 的输入参数	101
6.4.3	调用 Action	103
6.5	使用环境变量的参数化	105
6.5.1	定义和设置环境变量	105
6.5.2	在测试步骤中绑定环境变量值	106
6.5.3	导出环境变量到 XML 文件	108
6.5.4	导入外部环境变量文件	109
6.6	使用数据驱动器来参数化测试	110
6.6.1	数据驱动器的使用方法	110
6.6.2	数据驱动向导	111
6.7	文件数据源的数据驱动测试	113
6.7.1	文本文件作为数据源	113
6.7.2	Excel 作为数据源	117
第 7 章	关键字驱动测试	121
7.1	关键字驱动测试概述	121
7.2	关键字驱动测试方法	121

7.2.1	关键字测试驱动框架	122
7.2.2	关键字驱动测试的阶段划分	122
7.2.3	适用的测试阶段	122
7.3	关键字驱动测试步骤	123
7.3.1	编辑对象库	123
7.3.2	选择测试对象	123
7.3.3	选择测试对象相应的操作	124
7.3.4	设置测试逻辑	125
7.3.5	运行分析测试脚本	126
7.4	检查点的使用	127
7.4.1	为测试添加检查点	127
7.4.2	UFT 支持的检查点类型	127
7.4.3	添加标准检查点	128
7.4.4	添加图像检测点	130
7.4.5	添加图像检查点	131
7.4.6	选择对象	132
7.5	添加表格检查点	133
7.5.1	添加文本检查点	134
7.5.2	添加文本区域检查点	136
7.5.3	添加可访问性检查点	138
7.5.4	添加页面检查点	139
7.5.5	添加数据库检查点	142

第三部分 API 与服务测试篇

第 8 章	API/接口测试	146
8.1	API/接口测试的目的与意义	146
8.1.1	接口测试的目的	146
8.1.2	接口测试的意义	147
8.2	UFT 中的 API 测试	148
8.2.1	SOA 测试的重要性	148
8.2.2	SOA 概述	149
8.2.3	服务测试术语	149
8.3	API 测试通用流程	150

8.3.1	启动 API 服务	151
8.3.2	创建 API 测试	151
8.3.3	服务测试窗口	152
8.3.4	创建 API 测试步骤	153
8.3.5	连接测试步骤	157
8.3.6	多个数据源的数据映射	159
8.3.7	数据驱动	161
8.4	Web Service 测试	165
8.4.1	基于 WSDL 的 Web 服务测试	165
8.4.2	基于 REST 的 Web 服务测试	176

第四部分 非主流测试工具篇

第 9 章	Linux 系统交互应用功能测试框架工具 Expect	186
9.1	概述	186
9.1.1	TCL	187
9.1.2	Expect	188
9.2	适用场景	188
9.3	原理	189
9.4	使用介绍	190
9.4.1	Expect 的安装	190
9.4.2	简单例子	191
9.4.3	Expect 语法介绍	192
9.5	实战案例	192
9.5.1	自动输入密码	192
9.5.2	自动登录	193
9.5.3	sh 脚本中嵌入 Expect	194
9.5.4	自动登录 Ssh, Ftp	195
9.6	命令介绍	198
9.6.1	启动一个进程——spawn	198
9.6.2	expect	198
9.6.3	exp 命令	201
9.6.4	其他命令	203

第 10 章 网络协议自动化测试工具 Sulley	208
10.1 概述	208
10.2 适用场景	209
10.3 原理介绍	209
10.3.1 生成协议描述框架	209
10.3.2 模糊器处理	210
10.3.3 会话	210
10.4 使用介绍	210
10.4.1 安装 Sulley	210
10.4.2 Sulley 目录结构	211
10.4.3 Sulley 中的数据表示	213
10.4.4 Sulley 中的会话	220
10.4.5 Sulley 中的代理	221
10.5 实例分析	222
10.5.1 创建 FTP 协议描述框架	222
10.5.2 Sulley 会话	223
10.5.3 网络和进程监控	224
10.5.4 开始 Fuzzing 测试	225
第 11 章 无界面应用功能自动化测试工具 Peach	227
11.1 Peach 工具的概述	227
11.2 工具适用场景	228
11.3 工具原理	228
11.4 使用介绍	230
11.4.1 Peach pit 基本格式	230
11.4.2 数据间的依存关系的描述	233
11.5 实用案例	234
11.5.1 Zip 文件格式	235
11.5.2 Zip.xml 文件	237
11.6 命令附录	246

第一部分 理念篇

没有理论指导的实践是盲目的。

软件测试是一门实践学科，但软件测试绝不只停留在录制、回放等实践技巧上，熟悉并认知基础理论和理念是任何一个高阶软件人员都需要去把握和追寻的。

- 第 1 章 软件测试概论
- 第 2 章 数据驱动测试技术
- 第 3 章 关键字测试技术

第 1 章 软件测试概论

1.1 软件测试概论

1.1.1 软件测试概述

软件测试是软件开发整个过程中非常重要的步骤，是保证软件质量的重要环节。随着软件市场对软件质量要求的提高，国内外整个软件产业对软件测试的重视也与日俱增，软件测试在保障软件质量和改进软件开发工作中发挥着越来越大的作用。提高软件测试的准确率、软件测试时间预测，以及软件的管理过程也成了软件测试要解决的重要问题。

1. 软件测试定义

软件测试，描述一个用来促进鉴定软件的正确性、完整性、安全性和质量的过程。软件测试的经典定义是：在规定的条件下对程序进行操作，以发现程序错误，衡量软件质量，并对其是否能满足设计要求进行评估的过程（见图 1-1）。

软件测试是使用人工或者自动手段来运行或测试某个系统的过程，其目的在于检验它是否满足规定的需求或弄清预期结果与实际结果之间的差别。

它是帮助识别开发完成（中间或最终的版本）的计算机软件（整体或部分）的正确度（correctness）、完全度（completeness）和质量（quality）的软件过程；是 SQA（software quality assurance）的重要子域。

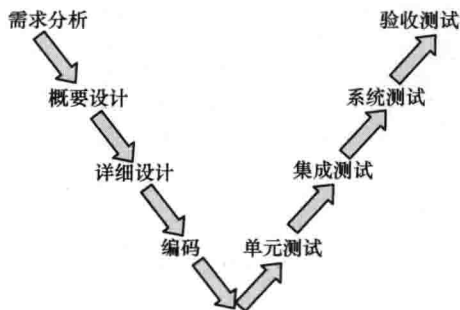


图 1-1 软件测试过程图

2. 软件测试的意义

软件测试在整个软件生命周期中占有非常重要地位，是保证软件质量的重要手段。鉴于软件质量的重要性，软件测试的重要性也得以彰显，因为哪怕微小的软件缺陷都可能在生产生活中造成严重的损失。从软件工程方面来讲，软件的质量是在整个软件开发过程中形成的。当软件程序代码完成之后，软件质量大体水平就基本确定了，虽然可以通过最后的确认测试发现大部分缺陷，但是，软件代码中存在的缺陷越多，遗漏的缺陷就会越多，质量在以后的修复中就很难得到质的改善与提高。

所以说来软件测试人员要尽早地发现缺陷与错误，督促和帮助开发人员修正问题。其原因很简单：实践表明，如果在需求阶段修正一个错误的代价是1个货币单位的话，那么，在设计阶段就是需求阶段代价的3~6倍，在编程阶段是需求阶段代价的10倍，在内部测试阶段是需求阶段代价的20~40倍，在外部测试阶段是需求阶段代价的30~70倍，而到了产品发布出去时，这个数字就可能是需求阶段代价的40~1000倍。软件测试的代价不仅很大，还有测试时间也比较长，有数据统计表明，软件测试一般会占到整个项目运行时间的40%左右，美国著名的软件工程专家Boehm的统计，软件开发总成本中，用在测试上的开销要占40%~50%，而对于可靠性要求高的医疗、航天等软件，软件测试的时间可能持续的会更久一些，一般至少要占软件开发周期的60%以上。软件在投入运行上线之前，要尽可能多的进行测试，以保证软件最终的运行质量。

1.1.2 国内外现状

目前，软件测试的相关研究一般集中在面向对象的软件测试方法研究，软件测试统计测试研究，需求驱动的软件安全缺陷自动测试生成等方面。由于软件复杂程度的提高，人们对软件测试的认识也不断提高，对于软件测试的重视度也越来越高。然而，总体来说软件测试研究的步伐还是落后于软件开发的。关于软件测试现在所面临的问题被称之为“杀虫剂悖论”，也曾形象地比喻成为“杀虫剂”。所以对软件系统进行测试以保障和提高系统正确性与安全性显得尤为重要。软件的开发过程中要不断对软件进行测试，为了保证软件的质量，测试过程的管理也非常重要，它可以进一步明确软件测试的每个细节，保证软件测试的每个环节不被遗漏。无论是一个小型的软件系统，还是一个大规模的系统，测试人员对于软件的测试都是非常重要的。因为软件的结构复杂程度决定了软件出现故障的可能性。