

Automated Software Testing
Introduction, Management, and Performance

国外IT精品丛书



软件自动化测试：引入、管理与实施

全面系统地介绍目前最有效的
自动测试工具、技术及方法

〔美〕 Elfriede Dustin
Jeff Rashka 著
John Paul

于秀山 胡兢玉 等译
张燕虹 审校


Addison
Wesley

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

Automated Software Testing
Introduction, Management, and Performance

软件自动化测试：引入、 管理与实施

Elfriede Dustin
〔美〕 Jeff Rashka 著
John Paul

于秀山 胡兢玉 等译

张燕虹 审校

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 提 要

本书详细介绍了软件自动化测试生存周期方法学所包含的自动测试计划、分析、设计、开发、执行和管理六个部分的内容,详细说明了如何对客户机/服务器和因特网应用进行自动化测试,并提供了许多软件自动化测试的最佳做法实例。

本书结构清晰,内容丰富,适用性强,适用于软件测试人员、软件开发人员、软件质量保证人员以及软件项目管理人员,也可以作为高等院校计算机专业师生的参考书。



Simplified Chinese edition Copyright © 2003 by Pearson Education North Asia Limited and Publishing House of Electronics Industry and Beijing Media Electronic Information Co. Ltd
Automated Software Testing Introduction, Management, and Performance by Elfriede Dustin, Jeff Rashka, John Paul, Copyright © 2002 All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison Wesley. This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macau).

本书中文简体字翻译版由电子工业出版社和Pearson Education培生教育出版北亚洲有限公司合作出版。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

本书封面贴有Pearson Education出版集团激光防伪标签,无标签者不得销售。

版权贸易合同登记号: 01-2002-5469

图书在版编目(CIP)数据

软件自动化测试: 引入、管理与实施/(美)·达斯廷(Dustin, E.)等著; 于秀山等译. —北京: 电子工业出版社, 2003.1

书名原文: Automated Software Testing Introduction, Management, and Performance
ISBN 7-5053-8290-X

I. 软… II. ①达… ②于… III. 软件—测试—自动化 IV. TP311.5

中国版本图书馆CIP数据核字(2002)第097892号

责任编辑: 贺玉寅

印 刷: 北京天竺颖华印刷厂

出版发行: 电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路173信箱 邮编: 100036

北京市海淀区翠微东里甲2号 邮编: 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 23.25 字数: 590千字

版 次: 2003年1月第1版 2003年1月第1次印刷

定 价: 40.00元

凡购买电子工业出版社的图书, 如有缺损问题, 请向购买书店调换, 若书店售缺, 请与本社发行部联系。联系电话: (010) 68279077

译者序

软件自动化测试是软件测试一个重要的组成部分，它能完成许多手工测试无法实现或难以实现的测试。正确、合理地实施自动化测试，能够快速、彻底地对软件进行测试，从而提高软件质量，节省经费，缩短产品发布周期。

软件自动化测试不同于一般意义上的软件测试，除了掌握一般的测试理论和方法外，还需要采用一些特殊的技术和方法。本书根据作者多年从事软件开发、软件自动化测试、软件管理方面的经验，从理论与实践的结合上全面、系统地阐述了如何开展软件自动化测试工作。本书详细介绍了软件自动化测试生存周期方法学六个组成部分的内容，详细说明了如何对客户机/服务器和因特网应用进行自动化测试，并提供了许多软件自动化测试最佳做法实例。本书可帮助软件测试专业技术人员和管理部门经济、有效、合理地开展软件自动化测试工作。

本书由于秀山、胡兢玉、赵红梅、贺玉寅、焦跃、杨玲萍、于乐山、霍亚青、李申、吴玲、张晓蓓、张灿、王成海、方志英、邓炜、罗维翻译，全书由张燕虹审校。

译者在翻译过程中，除对原文个别文字错误作了相应更正外，力求信、达、雅，但由于水平有限且时间仓促，错误与不妥之处在所难免，恳请广大读者谅解，并欢迎批评指正。

致 谢

特别感谢奥莉夫·约翰斯和克里斯·德赖尔对本书的热情支持。他们的鼓励、建设性的反馈和有价值的评论使本书内容增色不少。

感谢所有测试专业人员，如鲍里斯·贝泽在测试行业所发挥的领导作用以及对软件质量的执着追求。我们尤其感谢为本书出版做出贡献的下列所有人：

布拉德·阿普尔顿	布鲁斯·卡茨
斯泰西·科尼斯厄斯	柯克·诺思斯蔡尔德
马赛厄斯·戴格尔	马修·莱希
克里斯·德赖尔	蒂洛·林茨
乔伊斯·伊诺思	伊恩·朗
罗伯特·L·格拉斯	布雷特·舒克特
萨姆·古肯海默	罗伯特·舒尔茨
戴夫·古斯塔夫森	安迪·廷克哈姆
理查德·J·赫奇	威尔·特雷西兹
奥莉夫·约翰斯	克里斯·韦尔奇
安鲁拉达哈·卡雷	

前言

本书论述了当今软件专业人员所面临的挑战。这些挑战体现在实际进度、最终期限以及需要在项目中引入、管理和实施自动化测试方面。本书论述了客户机/服务器或因特网环境下的自动化测试。

本书的重点是论述那些需要完成更彻底和更快速测试的软件测试工程师/经理实际关心的事情、信息。出于同样原因，这些内容也适用于测试开发（即单元和组装测试）以及某些项目系统测试的软件开发责任人员。本书也为质量保证工程师对测试设计、测试程序和测试活动结果进行质量评审提供了有意义的指导。

负责整个开发工作的软件项目经理也可发现本书的价值。书中为项目经理提供了有关测试工作的目标以及进行自动化测试方面的指导，还提供了有关在某个项目中引入自动化测试的指导，并且概要论述了测试计划、设计、开发、执行和评估执行的过程。

本书作者与世界上许多自动化测试专业人员密切合作过。他们非常慷慨大方，愿意共享他们的问题和所关心的事情。这些测试工程师所表达的一个主要关注的事情与这样一个事实有关，即测试行业没有那种开发人员所享有的传统的结构化的方法论。与此类似，相对于传统的测试方法而言，项目经理、测试经理和测试工程师可能不熟悉进行自动化测试所需要的各种方法。

显然，强调自动化测试是软件行业变化的一个有代表性的范例。这个变化不能简单地概括为工具的使用以及测试自动化的效果。相反，它遍布整个测试生存周期和系统开发生存周期。其结果是，项目经理、测试经理、软件工程师和测试工程师所采用的方法被改变了。为使软件专业人员成功地跨越到自动测试，必须使他们接受结构化的测试方法。

本书的创新性在于它传播了一个适用于整个测试生存周期的新的结构化的模块方法，同时也提供了业内专业测试人员需要的相应的测试自动化和测试管理方面的指导。

测试自动化

制作应用程序的软件项目经理和软件开发人员面临着进度不断缩短、使用的资源最小的挑战。作为他们少投入多产出尝试的一部分，企业想充分地测试软件，但要求尽可能地快和彻底。为了实现这个目标，于是企业转向了自动测试。

面对这个现实并认识到很多测试不能手工执行，例如，模拟1000个虚拟用户的容量测试，软件专业人员正在将自动测试引入到他们的项目中。尽管需要引入自动测试，软件专业人员可能不知道在软件项目中引入自动测试究竟包括哪些内容，他们也可能不熟悉当今自动测试工具应用的广度，本书在这些领域提供指导。

自动测试能力的增长很大程度上起源于快速应用开发（RAD）普及的增长。快速应用开发是一种软件开发方法学，其重点在于通过增量的方式提供常见的软件构件将开发进度缩短到最少。RAD的目的是保证用户尽早设计和开发每一个构件，以便于细化软件，从而确保它更准确地反映用户的需求。在这个贯穿每一个软件构件始终，软件频繁变化和增加的环境

里，软件需求被迫改变，软件测试呈现出本质上的重复。经过相当数量的新测试以及现有测试脚本的返工（就像先前发布的软件模块有时需要返工一样），完成了每一个新构件。假设软件应用程序要发生频繁的变化和增加，那么，自动软件测试就成为确保软件每一个构件准确、稳定的一个重要的控制机制。

如上所述，RAD的一个主要目标是通过阐述早期构件开发中最有风险的部分以缩短整个开发进度，从而能够在初始RAD周期开始阶段以及贯穿每个子RAD周期就开展测试活动。如本书第三部分所述，测试设计和开发是一项复杂的任务。事实上，测试工作可能像开发软件应用程序一样耗时。例如，当项目包含与流行的商业产品集成时，测试工作需要的资源可能比软件开发需要的多。当测试小组没有参加软件规格说明编制工作或没有一开始就开展测试工作时，就给项目带来了风险。在这些情况下，其潜在的后果包括不完善的测试工作、不富余的测试进度和对测试进度做计划外的扩充以适应测试工作。

现在，测试一个项目需要的大量工作需要自动测试工具的支持。手工测试是一个劳动密集型工作，并且容易出错。它不支持那些可能由自动测试工具完成的相同种类的质量检查。引入自动测试工具能够用更有效、可重复的自动测试环境代替世俗的手工测试活动。这项工作本身提高了测试工程师的士气和持久力。

尽管有些自动测试工具始于捕获和回放工具，但测试工具套件的功能和能力一直都在扩展。对软件产品自动测试的能力包括测试图形用户界面、需求一致性、负载测试、代码覆盖、因特网接口、网络通信、内存泄漏等。新的能力在继续增加，以保持与不断增长的测试支持需求同步。

ATLM——自动测试生存周期方法学

本书致力于自动测试生存周期方法学（ATLM）范围内软件测试专业人员关心的事情。自动测试生存周期方法学是一个旨在确保自动化测试成功实施的结构化的方法学。自动测试生存周期方法学方法反映了现代化的快速应用开发工作的益处。在这个过程中，在开发周期的早期就聘请用户参加这些工作。软件产品的最终用户主动参与每一个软件构件的分析、设计、开发和测试工作，该产品是采用增量模式而增加的。

自动测试生存周期方法学包括了一个多个阶段的过程，该方法学由六个部分组成。它支持判断是否获得一个自动测试工具所必需的细节和相关的活动。该方法学考虑了引入和优化一个自动测试工具的过程，并论述了测试计划、分析、设计、开发、执行和管理。在测试计划中，概要描述测试的范围，该测试范围是对测试方法和实现的顶层描述。通过定义测试目标、目的和策略以及测试需求进一步缩小测试范围。与软件应用开发类似，测试需求要在构造测试设计之前确定。同样，必须制订测试大纲并对其精心设计以确保对目标应用程序进行最经济、最有效的测试。采用测试工作图描述的方法来开发测试设计，从而使项目和测试人员对测试大纲的边界和范围有一个大致的了解。

测试培训

自动测试的发展为软件工程师带来了选择新职业的机会。事实上，当自动化的软件测试专业人员需求爆炸时，社会上的学院和大学还没能做出相应的响应以满足行业的需求。

目前，大学、公司和政府运营部门已经在积极主动地响应软件行业发生的这些变化并开设了软件测试和质量保证课程。例如，乔治梅森大学（GMU）开设了软件测试和质量保证课程（为获取更多的信息，可访问GMU因特网网站<http://www.isse.gmu.edu/>）。堪萨斯州大学（KSU）为学生开设了几门有关软件测试和质量保证以及其他涉及该学科的课程（见KSU因特网网站<http://www.ksu.edu/>）。

珀杜大学开设了两门软件工程方面的大学课程，该课程覆盖软件测试和可靠性两方面的内容。珀杜还建立了一个软件工程研究中心（<http://www.uoregon.edu/serc/>）。来自8个大学的教员和11个公司的代表参与了该中心的工作。除了其他领域外，该中心支持软件质量方面的研究。为获取更多的信息，可访问珀杜大学因特网网站<http://www.purdue.edu/>。

北西雅图社区学院建立了美国最现代的测试课程之一。学院开设了三个等级的软件测试课程（入门、自动化和领导才能）以及1年和2年的软件测试程序项目。关于这些课程和项目更多的信息，可访问网站<http://nscxux.sccd.ctc.edu/>。其他的大学和行业培训课程方面的信息，请访问作者的网站<http://www.autotestco.com/>。有关公司和其他测试培训组织，见附录C中的表C.3。

本书旨在为使用现代化的自动测试工具进行软件测试提供室内指导。本书为学生提供了应用入门和软件测试重要性方面的内容，并描述了能够实施的不同种类的自动测试。本书还对测试方法定义、测试小组的任务和职责、测试计划、测试设计、测试脚本开发、测试执行、缺陷跟踪和测试进展报告进行了描述。

作者简介

本书由软件行业的下述三名专业人员合著而成。

埃尔弗里德·达斯廷是一名计算机系统分析员/程序员，她使用CASE工具开发过软件应用程序与实用程序、过程和数据建模以及系统设计模拟模型。她具有支持各种系统应用开发项目的经历，这些项目涉及保健、财务、后勤和企业信息管理系统。此外，埃尔弗里德负责过整个开发生存周期的实施，从需求分析、设计、开发到软件自动化测试。她是一名测试经理、首席测试顾问，对许多项目的自动测试进行了指导。由于她具有自动测试方面的专业知识，人们请她帮助改进商用测试工具产品的能力，其中，她使用这些测试产品所提供的反馈意见是非常宝贵的。

杰夫·拉什卡管理过许多重大信息系统与系统集成项目。他担任过一些系统应用项目经理，这些系统应用有世界运输资产管理、企业信息管理、财务管理、条形码清单管理以及舰载信息系统。杰夫在履行软件工程研究所的能力成熟度模型（CMM）内的准则方面具有过程改进管理经验。

约翰·保罗是从事财务和预算系统以及其他主机结构信息系统的高级程序员/分析员。他领导过用许多不同方法与编程技术进行系统分析与设计、应用原型以及应用开发方面的软件开发。他负责过使用自动测试工具进行应用程序测试的开发。约翰还担任过2000年问题依从性测试的领导角色。

三位作者用他们的软件工程技术、自动测试与管理方面的知识编成了这本书。本书论述了软件测试工程师和经理须知的事项与信息。对于那些负责软件测试活动的软件工程师与软件项目经理来说，本书是一本实用的入门书。

本书风格与编排结构

本书的编排结构与自动测试生存周期方法的阶段、任务以及步骤有关。对本书的排序方式也做了特意安排。读者拿着它就好像拿着一本手册，书中教你如何完成某一项目的自动软件测试。第一个问题是：自动测试究竟是什么？为什么我需要它？本书第一部分予以解答，同时，还给出了其他一些基本常识，使即将成为测试工程师的读者满怀信心地接触这个新领域。之后，本书指导读者进行自动测试、自动测试工具选择和评估。

读完本介绍以后，测试工程师会有另外一些问题：配备工具涉及哪些问题？如何将测试小组放在适当的位置？需要什么样的初期测试计划？本书第二部分回答了这些问题。尤其是简要叙述了自动测试工具的引入过程，另外还给出了测试小组的构成准则。

本书第三部分着眼于自动测试计划、分析、设计和测试自动化（编程）。这一部分讲述了测试设计技术，该技术可与20年前的结构化软件设计技术媲美。特别是它突出了测试自动化设计中所要求的规范。目的是给出测试设计和测试用例开发方面的有用信息，这样一来，测试工程师不必去探讨（通过试验和错误）如何将良好的测试设计和测试程序放在一起这个问题。

本书第四部分帮助读者论述其他一些问题：执行测试中涉及哪些问题？如何管理测试进度？如何记录和跟踪缺陷？本部分提供有关测试执行、缺陷跟踪以及测试程序状态跟踪方面的准则。开发和执行自动测试过程的一套最佳做法已通过有效的方式提供给执行测试的测试专业人员。

总之，本书力图减少自动测试生存周期方法学框架内软件测试专业人员所关心的事项。自动测试生存周期方法学是一种结构化的方法，它可以确保成功执行自动测试。读者如有问题或意见，请与作者联系，网址：<http://www.autotestco.com/>。该网址还将提供更多有关自动软件测试与资源方面的信息。

目 录

译者序	v
致谢	vi
前言	vii
第一部分 何谓自动测试	1
第1章 自动测试的产生与发展	1
1.1 自动测试	1
1.2 软件测试的背景	2
1.3 自动测试生存周期方法学	3
1.4 自动测试生存周期方法学在软件测试领域中的作用	8
1.5 软件测试职业	13
本章小结	16
参考文献	16
第2章 自动测试决定	18
2.1 克服不正确的自动测试期望	19
2.2 自动测试的好处	23
2.3 获得管理部门支持	34
本章小结	41
参考文献	42
第3章 自动化测试工具评价和选择	43
3.1 企业的系统工程环境	45
3.2 适用于测试生存周期的工具	48
3.3 测试工具调查	57
3.4 评估范围定义	62
3.5 亲手评估工具	64
本章小结	66
参考文献	67
第二部分 项目中引入自动测试	69
第4章 自动测试引入过程	69
4.1 测试过程分析	71
4.2 测试工具考察	86

本章小结	93
参考文献	93
第5章 测试组管理	95
5.1 测试组的组织结构	96
5.2 测试任务	101
5.3 测试工作规模	106
5.4 招聘测试工程师	111
5.5 任务及职责	117
本章小结	120
参考文献	121
第三部分 测试计划与准备	123
第6章 测试计划：灵活运用测试	123
6.1 测试计划活动	124
6.2 测试范围	127
6.3 测试需求管理	132
6.4 测试事件、活动和文档	135
6.5 测试环境	137
6.6 测试计划	138
本章小结	141
参考文献	142
第7章 测试分析与设计	143
7.1 测试需求分析	144
7.2 测试设计	149
7.3 测试程序设计	163
本章小结	180
参考资料	182
第8章 测试开发	183
8.1 测试开发体系结构	185
8.2 测试开发指南	197
8.3 自动测试基础设施	217
本章小结	223
参考文献	224

第四部分 测试执行和评审	225
第9章 测试执行	225
9.1 执行和评估测试阶段	226
9.2 缺陷跟踪和修改过程	233
9.3 测试活动状态跟踪	236
本章小结	245
参考文献	245
第10章 测试活动评审与评估	246
10.1 测试活动中的经验教训——改正和改进的措施	247
10.2 测试活动的投资回报	254
本章小结	260
参考文献	261
第五部分 附录	263
附录A 如何测试需求	263
附录B 支持自动测试生存周期的工具	272
附录C 测试工程师培养	313
附录D 测试计划实例	322
附录E 最佳做法	352

第一部分 何谓自动测试

第1章 自动测试的产生与发展

一项包含软件测试自动化的有效测试本身就是一个小型的开发生存周期。自动测试相当于一项与策略、目标规划、测试需求定义、分析、设计、开发、执行和评估活动有关的工作。

1.1 自动测试

“我们马上需要新的软件应用程序。”“我们现在需要那些新产品特性。”听起来是否熟悉？

人们要求当今的软件经理和开发人员在总是紧缩的进度内用最少的资源彻底改变其产品。90%以上的开发人员耽误了交货日期。67%的开发人员常常错过最后期限。此外，91%的开发人员在开发周期后期不得不取消关键功能以赶上最后期限[1]。斯坦迪什集团报告提供了类似的结论[2]。产品投放市场的早晚可能造成产品的生死存亡以及公司的生死存亡。

业务和政府机构也面临着降低其成本的压力。解决这个问题的主要途径之一是通过软件应用程序来实现业务处理的进一步自动化和流水线化。负责应用程序开发工作的业务经理和政府领导不愿意等待一年或更长时间才看到能够运行的产品，相反，他们正在拟订软件开发工作规范，这些开发工作侧重于缩短开发进度，这时常需要采用增量方法构建软件。尽管这些增量的软件版本为用户查看和使用提供某些有形的东西，但是，将某一软件构件版本与下一软件构件的后续版本结合的需求将增加测试工作的强度与复杂性。

一些企业在尝试低耗高效工作时打算在短的进度内充分测试其软件。为了达到这一目标，这些企业转向自动测试。自动测试的一般定义为：各种测试活动的管理与实施，包括测试脚本的开发与执行，以便使用一种自动测试工具来验证测试需求。测试活动自动化在许多情况下可提供其最大价值，如测试脚本被重复的地方或测试脚本子程序被生成而后被许多测试脚本重复调用的地方。在开发和组装阶段（其中可重用脚本可能运行很多次），这种测试有很高的回报。

使用自动测试工具对后续的增量软件构件进行综合测试提供了巨大价值。每一个新的构件都引发相当多的新测试，而且可重用先前开发的测试脚本。假定需求和软件不断变更和增加，那么，自动软件测试则作为一个重要的控制机制，确保软件每次重构的准确性与稳定性。

系统测试级的回归测试是有效应用自动测试的另一个例子。回归测试设法验证改进后的系统或软件产品提供的功能是按照规定执行的，在系统或产品运行中没有出现非预期变化。自动测试便于回归测试以非常有效的方式执行（本书介绍了一些自动测试效率方面的细节和例子）。

为了了解自动测试的来龙去脉，有必要介绍一下测试种类，测试工作一般是在各种应用程序开发生存周期阶段进行的。在客户机/服务器或因特网环境下，目标系统所涉及的不止一个软件应用程序。的确，测试可能跨越多个平台，涉及支持应用程序的多个层，涉及与许多商业现成产品的接口，利用一个或多个不同类型的数据库，涉及前端与后端处理。这种环境下的测试可能包括功能需求测试、服务器性能测试、用户界面测试、单元测试、组装测试、程序模块复杂性分析、程序代码覆盖测试、系统负载性能测试、边界测试，安全性测试、内存泄漏测试以及其他许多类型的评估。

自动测试目前可支持这些类型的测试，因为最近几年自动测试工具的功能与能力已经扩展。这种测试以更有效且可重复的方式（与人工方式相比）进行。自动测试能力不断提高，以便跟上应用程序生产更快、更好且更低廉的不断发展的步伐。

1.2 软件测试的背景

软件测试的历史反映了软件开发自身的发展演变。长期以来，软件开发侧重于在大型或小型计算机平台上开发的与公司数据库系统有关联的大型科学计划与国防部计划。在这一阶段，测试方案都写在文件上。测试工作的目标是控制流路径、复杂算法的计算以及数据操作。一组有限的测试程序可有效地测试整个系统。测试工作一般在项目进度的最后时刻才着手进行，最后时刻的测试工作是由当时最合适的人士完成的。

个人计算机的出现促进了整个工业界的标准化，现在开发的软件应用程序要在通用操作系统上运行。个人计算机的引入产生了一个新的阶段，并引发了商用软件开发的爆炸性发展。在这种情况下，商用软件激烈竞争以求得地位和生存。在产品市场上适者生存，计算机用户将生存下来的软件当作事实上的标准。逐步壮大的联机系统，替代了批处理方式操作。因为几乎可以用任何方式调用联机系统中的作业流，所以联机系统的测试工作要求以不同的方式来进行测试设计。

客户机/服务器体系结构利用了合适专业的软件，这是通过使用前端图形用户界面应用程序开发工具和后端数据库管理系统应用程序以及利用联网的个人计算机的普遍可用性实现的。术语“客户机/服务器”说明了两个软件处理流程之间的关系。其中，客户计算机请求一个由服务器计算机执行的服务。一旦服务器完成了所要求的功能，它便向客户机发回结果。尽管客户机和服务器可在一个单一的机器上运行，但通常在由网络连接的分散的计算机上运行。图1.1提供了客户机/服务器体系结构的示意图形。

客户机/服务器应用的流行给测试工作带来了新的复杂局面。测试工程师不再像过去那样使用在单一系统上工作的单一、封闭式应用程序。相反，客户机/服务器体系结构涉及3个不同的部分：服务器、客户机和网络。平台间连接增加了出错的可能性，因为迄今为止几乎没有制订出客户机/服务器标准。因此，测试牵涉到服务器与网络的性能以及跨越3个部分的整个系统的性能和功能。

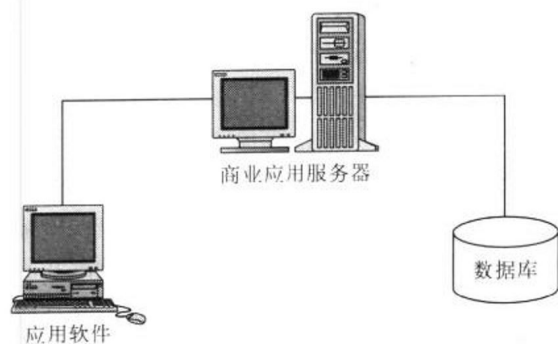


图1.1 客户机/服务器体系结构

伴随着客户机/服务器体系结构带来的新的复杂局面，图形用户界面屏幕的特性带来了另外一些挑战。图形用户界面替代了基于字符的应用，并通过减少用户对指令了解的详细程度或软件功能背景的程度从而使几乎任何人都可以使用软件。图形用户界面涉及用户屏幕窗口内的信息表示。

用户屏幕视窗包含可被选用的对象，因此可使用户控制其逻辑。这些屏幕能将对象表示为图标，屏幕似乎能无穷尽地变幻。例如，屏幕图像大小可变，监视器屏幕上的屏幕图像位置可以改变，能以任何方式在任何时间选择任何对象，对象可以改变位置。这种方法称为事件驱动环境，它与基于主机的过程环境截然不同。

考虑到客户机/服务器应用的性质，就选择对象的方法而论，存在着相当大的随意性。一般来说，在应用程序内部没有明确的路径，可以用不同的顺序选择对象。相反，通过一定数量的路径调用和执行模块其结果是下面一种情况，测试程序不容易执行所有可能的功能。因此，测试工程师必须将其测试活动侧重于实现大多数系统需求的应用程序部分和用户可能使用系统的方法上。

随着图形用户界面应用的普及，采用屏幕捕捉与回放的方法测试应用程序受到人们的关注。自动测试工具具有这种能力，它们被投放市场以满足这种需求并逐渐产生势头。尽管测试方案和脚本通常仍使用字处理应用程序编写，不过，自动测试工具的使用增多了。更复杂的测试工作要求更庞大且更详尽的规划。要求实施测试的人员更熟悉被测试的应用程序，并且具有与平台和网络有关的更专业的技能要求（也适于正在使用的自动测试工具）。

自那以后，支持屏幕捕捉和回放的自动测试工具日益成熟并且能力也增强了。具有各种特性的自动测试工具不断涌现。此外，自动软件测试日益成为一种编程行为，尽管它不断涉及传统的测试管理功能，如需求跟踪能力、测试计划、测试设计、测试场景和脚本开发。

1.3 自动测试生存周期方法学

现已证明，使用自动测试工具支持测试过程有助于提高产品质量并且使项目进度和工作量减少到最小（参阅第2章中的“案例研究：测试自动化测量的价值”）。为了获得这些好处，必须在项目初期就着手测试活动和测试计划。因此，在业务分析和需求活动中，必须吸纳测试工程师参加，并且测试工程师应参与分析和设计评审活动。这些评审内容可作为有

效的测试方法，从而避免以后出现分析/设计错误。这种早期的参与可确保测试组了解用户的需求，这将有助于制订可满足适当测试环境的体系结构并产生更详尽的测试设计。

早期的测试参与不仅确保有效的测试设计（当利用自动测试工具时，这是一项必不可少的重要活动），而且能够早期检测错误并且避免错误由需求规格说明转移到设计，再从设计进入代码。这种避免措施降低了成本，减少了返工并节省了时间。在开发周期中，错误被发现得越早，它们则容易被纠正且成本低。成本是根据排除错误所需的时间量和资源来测量的。初期阶段发现的故障纠正起来相对容易，对运行没有影响，几乎不需要资源。相反，运行阶段发现的缺陷涉及若干部门，可能要求广泛的重复测试，并且可能造成运行的停止。表1.1列出了整个开发生存周期不同阶段缺陷检测的开销情况[3]。

表1.1 错误预防比处置错误更划算

缺陷消除费用随系统开发生存周期而倍增	
阶段	成本
定义	1美元
概要设计	2美元
详细设计	5美元
编码	10美元
单元测试	15美元
组装测试	22美元
系统测试	50美元
交付后	100美元

本书通篇做了讨论并在图1.2示出的自动测试生存周期方法学代表了实施自动测试的结构化方法。自动测试生存周期方法学反映了现代快速应用程序开发工作的好处，在此情况下，这些工作吸引了用户早期参与每一个采用增量方式构建的软件版本的分析、设计和开发工作。

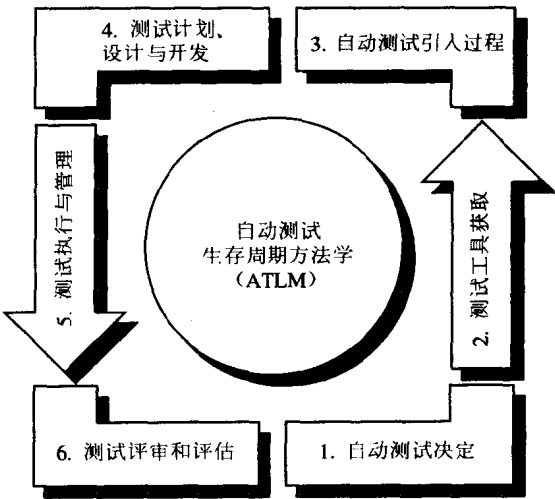


图1.2 自动测试生存周期方法学

为了遵循自动测试生存周期方法学，测试工程师在系统生存周期的早期就参与工作，从业务分析到整个需求阶段以及每一软件构件的设计和开发阶段。早期参与可使测试组对需求规格说明和软件设计进行详尽的评审，更全面地了解业务需要和需求，设计最适当的测试环境并产生更严格的测试设计。使用测试方法学（如与开发生存周期并行的自动测试生存周期方法学）的一个次要的好处是在软件开发人员与测试工程师之间形成了紧密的工作关系，这促进了单元测试、组装测试和系统测试期间的合作并产生更好的成果。

参与早期测试意义重大，因为需求或使用用例是确定测试需求和判定测试成功与否的基础或基准点。系统或应用程序的功能规格说明应由测试组评审。准确地说，至少必须使用下文给出的标准（详见附录A）评估功能规格说明。

- 完整性。评估需求被详尽定义的程度。
- 一致性。确保每一需求都不会与其他需求相矛盾。
- 可行性。评估通过可用技术、硬件规范、项目预算与进度以及项目人员技能等手段实际实现需求的程度。
- 可测试性。评估测试方法可证明需求已被成功实现的程度。

应在功能规范/需求阶段确定测试策略。能够用于需求阶段的自动工具可有助于产生可测试的功能需求，因而减少了测试工作并降低了测试成本。考虑到测试自动化，产品设计和编码标准可提供合适的环境以最大限度地使用测试工具。例如，开发工程师可把可测试性设计和构建技术应用到应用程序代码中。第4章进一步讲述了可测试代码的构建。

采用自动测试生存周期方法以支持涉及自动测试工具的测试工作，自动测试生存周期方法学包括多级过程。该方法支持各种详细与相互关联的活动（这些活动用于决定是否使用自动测试工具）。它研究了引入和利用自动测试工具所需的过程，涉及测试开发与测试设计以及测试执行与管理。该方法也支持测试数据以及测试环境的开发与管理，并描述制订测试文件的编写方法，以便形成各种问题报告。自动测试生存周期方法学提出了一种可描述某一过程（根据这一过程处理和执行测试）的结构化的方法。这种结构化方法对于帮助指导测试组避免犯下面一些常见的测试错误是必要的：

- 实施测试时，使用自动测试工具，而无合适的测试过程，结果产生了一个特定的、不可重复的、不可测量的测试程序。
- 实施一项测试设计时，不遵循任何设计标准，结果生成了不可重复的测试脚本，因而增量的软件构件不可重用。
- 当正在应用的工具不能将所有测试自动化时，试图使测试需求100%的自动化。
- 使用错误的工具。
- 在应用程序开发周期中启用测试工具太晚，没能将充足的时间用于工具建立和工具引入（即没有提供学习曲线）。
- 测试工程师参与应用开发生存周期太晚，结果是不能很好地了解应用和系统设计，因而无法完成测试。

自动测试生存周期方法学旨在确保自动测试的成功实施。如表1.2所示，自动测试生存周期方法学包括6个主要过程即6个主要组成部分。每一主要过程进一步分成若干从属过程。