

igned for
rosoft®
JowsNT®
Jows®98

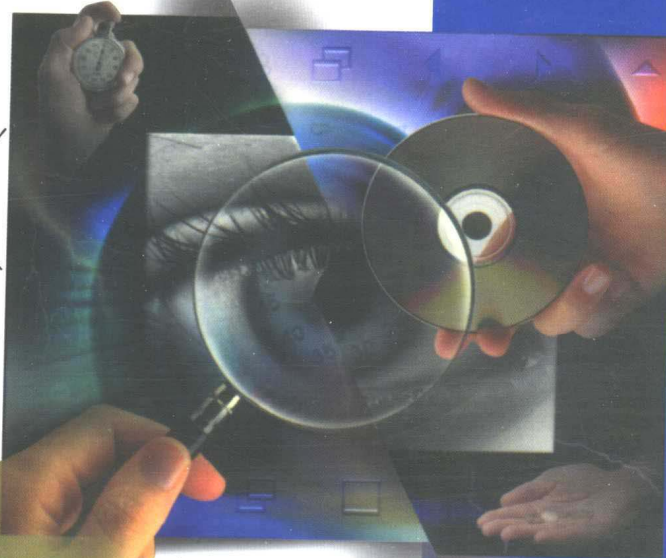


D-ROM
cluded

测试流程管理

Microsoft Press

管理硬件和软件测试项目
的实践工具和技术



北京大学出版社

测试流程管理

[美] Rex Black 著

天宏工作室 译

北京大学出版社

·北京·

著作权合同登记 图字：01-1999-1924 号

Managing the Testing Process

Rex Black

本书版权为 Rex Black 所有,1999。(Copyright © 1999 by Rex Black. All rights reserved.)

本书中文版由美国 Microsoft 出版社授权北京大学出版社独家出版,2001。

本书封面贴有北京大学出版社的激光防伪标签,无标签者不得销售。

版权所有,翻印必究。

图书在版编目(CIP)数据

测试流程管理/(美)布莱克(Black, B.)著;天宏工作室译. —北京:北京大学出版社, 2001.3

ISBN 7-301-04177-2

I. 测… II. ①布… ②天… III. ①电子计算机-测试 ②软件-测试 IV. TP306

中国版本图书馆 CIP 数据核字(2000)第 71768 号

书 名: 测试流程管理

著作责任者: [美]Rex Black 著 天宏工作室 译

责任编辑: 邱淑清 范 彦

标准书号: ISBN 7-301-04177-2/TP·453

出版者: 北京大学出版社

地 址: 北京市海淀区中关村北京大学校内 100871

网 址: <http://cbs.pku.edu.cn>

电 话: 出版部 62752015 发行部 62754140 编辑室 62757065

电 子 信 箱: zpup@pup.pku.edu.cn

排 印 者: 北京大学印刷厂

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 毫米×1092 毫米 16 开本 19.5 印张 390 千字

2001 年 3 月第 1 版 2001 年 3 月第 1 次印刷

定 价: 52.00 元(含光盘)

致谢

我首先要感谢 Microsoft 出版社的同事,他们帮助我将自己关于测试流程管理的思想整理为能够对您有实际帮助的一本书。特别感谢 Mary Renaud、Wendy Zucker、Ben Ryan 和 Erin O'Connor。我要感谢三位优秀的专家,他们审阅了本书的原稿,使我从中受益颇丰。Dr. Boris Beizer、Reynolds MacNary 和 Jay Nelson 向他们的批评家展示了自己特殊的才能、独到的见解和广阔的阅历。

在学习如何管理测试项目的过程中,许多天才的专家与我一同工作。冒着遗漏某些人的危险,我要特别感谢多年来帮助我的以下人员(以随机顺序):

Dr. Shahram Zaman、Dr. Art Gooding、Dr. Richard Baker、Kyle Marple、Tom Bohner、Jim Magdich、Troy Sukert、Bill Schoneman、Bob Petersen、Jeff Fields、Dave Desormeau、Jerry MacMillan、Armand Aghabegian、Bob Hosale、Neal Pollack、Chris Baker、Greg Cummings、Shawn Panchacharam、Summer Chien、Dr. Bertrand Chen、Kefetew Selassie、Craig Rittenhouse、Terry Newgard、Gary Hochron、Greg Scala、Dr. Alonzo Cardenas, 以及 Dr. Shoichi Minagawa 等。

当然,我非常感激过去和现在的所有同事、转包商、雇员、客户和雇主等。

除此之外,我还要感谢我的父母 Rex Sr. 和 Carolynn,感谢他们对我多年的爱护和支持。对本书最重要的是我的妻子和商业伙伴 Laurel Becker。编写本书使我忽略了许多正常生活,但我特别感激我的妻子牺牲了她自己的时间来帮助我。本书应该献给她!

请将本书中的所有错误、遗漏、过失、意见和糟糕的笑话归结于我。

简介

您的职责是管理计算机硬件或软件的测试项目吗？恭喜！您可能刚刚从测试工程师晋升为管理人员，或者刚刚从另一个开发小组调过来，也可能您已经做过一段时间的测试项目。无论您是一位测试管理者、开发管理者、技术或项目领导，还是对公司的测试和质量保证程序有一定责任的个体贡献者，您都有可能正在寻找有关如何管理测试这种独特项目的方法。

本书包含了我希望在从程序设计和系统管理转移到测试管理时需要了解的知识。它将向您展示如何开发一些关键的工具，并将它们应用到测试项目中，它还提供了一些技术来帮助您获得成功所需的资源。如果您掌握了基本的工具，应用技术来管理资源，并适当注意每一个领域，那么您就可以成功地管理测试项目。您甚至可能会做出出色的成绩，这有可能使您成为像我这样的终身的测试项目管理者。读不读本书完全由您自己决定。

本书的重点

编写《测试流程管理》有几个原因。首先，在某些人称为的“软件危机”中，在论及交付日期和质量时，许多项目在期望与现实之间都存在着差异——创建和测试软件的个体贡献者、高级项目管理者，以及客户都有过这样的经历。同样，计算机硬件开发项目常常会错过关键的时间计划和质量标准。作为项目风险管理的一个集成部分，有效测试和清楚的交流可能会对此有帮助。

其次，我察觉到软件和硬件测试文章中的差异。我曾经读到过介绍如何设计和实现测试实例的初级问题的书，以及告诉复杂项目的管理者如何使用诸如 ISO 9000、“总体质量管理”、软件质量规范等概念和工具来将他们的产品提高到更高质量水平的书。但是，我认为像我们这样的测试管理者需要一本介绍测试项目管理的基本工具和技术，这是主要的原材料。

本书提供的提示和工具将帮助您计划、建立和执行结构化测试操作。与常见

的、但常常会起负面作用的测试项目相比,结构化测试操作是经过规划、可重复的,并且能够归档的过程。本书介绍的知识将使您能够开发一些模型来理解测试所生成的大量数据点的意义,以便您能够有效地管理经常令人迷惑的、混乱的、多变的软件或硬件开发项目。它还将向您展示如何建立有效而经济的测试组织。

最后,我将开发环境中的测试管理所特有的主题选为本书的重点。因为其他书中已经很好地介绍了下面两个相关的主题,所以这里我不再过多地介绍它们:

- ◆ 基本的项目管理工具,如工作细目分类结构、甘特图、状态报告,以及人事管理技巧等。这些工具无疑是您需要掌握的重要部分。
- ◆ 计算机硬件“生产”测试。如果您涉及到这类测试,那么我建议将 W. Edwards Deming、Kaoru Ishikawa 和 J. M. Juran 编写的书作为有关统计质量控制的优秀资源,并将 Patrick O'Connor 的书作为有关可靠性工程的优秀资源。

在将不变的“黄金代码”复制到发布媒体这一意义上,软件产品不需要测试。但是,也有这种情况,硬件和软件生产常常包括小的修订和维护版本。您可以使用本书介绍的技术来管理涉及这类版本的小型测试项目。

与硬件相反,本书详细介绍了测试软件的细节,这可能使本书初看起来好像是沿着两个方向进展。不过,我发现,对于这两个测试领域的差异,从“测试项目”不如从“测试方法”的角度来看重要。这是合情合理的:硬件测试软件,同时软件也测试硬件。这样,您就可以使用相同的技术来管理硬件和软件开发项目中的测试工作。

规则还是菜谱?

当我最初作为测试工程师和测试项目管理者时,我对测试还是一窍不通。当然,无知常常会导致您意识不到对面射来的隧道光线实际上是即将驶来的列车。我想,“测试能会有多难呢?测试只不过是想象可能会出现什么问题,然后试着验证这些问题罢了。”

不过,我很快就发现,这个推测存在着三个主要问题:

- ◆ “想象可能会出现什么问题,并试着验证这些问题”所涉及的任务(也就是说,设计优秀的测试实例)实际上是非常困难的。比我更聪明的人已经编写了更好的书来介绍测试实例工程的艺术性。不幸的是,我的教授并没有向 Boris Beizer 或 Glenford Myers 等人引见我和我的尚未毕业的主修计算机科学的同事。在我还是一名学生时,我就曾阅读过几本有关软件工程的书,它们最多用几页篇幅来介绍测试过程。或许这也正说明了我为什么不能理解撰写优秀的测试实例的

困难。它也可以对计算机产业中的艺术状态产生的许多好奇做出解释。

- ◆ 测试并不是在真空中进行的。相反,它是开发项目整体的一部分——因此,测试必须适应真正的项目需要,而不是像黑客那样摆弄计算机以查看能破坏些什么。总之,测试项目需要进行测试项目管理。
- ◆ “测试能会有多难”这种思想的盛行只是增加了测试专家面临的困难。一旦我们通过痛苦的经历确切地知道测试的真正困难时,有时就会感觉我们命中注定(就像 Sisyphus 和 Dilbert 之间的十字架一样)要一遍又一遍、一个项目接着一个项目地解释,为什么这个测试项目花了如此长的时间和如此多的资金。

这几点隐含了几个复杂的因素。其中,最重要的一点是,组织中测试过程的成熟程度有很大的不同:测试可能被认为是令人讨厌的事后诸葛,也可能是交付优质产品的关键步骤。此外,激励因素(为什么要进行测试管理的原因)在重点和强度上都可能有所不同。我知道,被极大的恐惧所激励的管理者与想要生产出优质产品的管理者对测试的看法是不同的。最后,测试从业者自己的设备主要是学术界留下的,他们被迫通过反复的试验和失败来创建程序,从而导致项目管理过程中仍存在着很大的可变性。

这些因素使得开发一个有关“如何”计划和执行测试项目的指南变得十分困难。学术界的人士可能会说,测试项目管理者不愿意参与简单的规则开发。“理解以下思想,您就能理解这一领域。”这句话不适用于测试领域。测试规则的开发当然不是我要在本书中介绍的内容。

但是,您需要正确管理测试项目的规则吗?我认为不需要。例如,假设我是一位有能力和多才多艺的业余厨师。我将永远不会出现在世界知名厨师的行列中,但是我常常会为我的家人做出精美的晚餐。我曾多次成功地准备了丰盛的感恩节晚餐,其中一些是在旅馆的厨房中做出的。当我在大学学习期间,我掌握了如何在一定的条件下做出尽可能好吃的菜。在这一过程中,我学会了如何阅读菜谱上的烹饪方法,将它们应用到现实需要中,适当增减一些原料,以解决从正规晚餐到简单快餐的时间问题,并根据情况随机应变。

在一定的条件下做出好吃的菜是管理部门希望从测试组织获得的收益的一个很好的比喻。于是,本书应该作为测试项目管理者的“菜谱”,向您介绍所需的基本工具,并帮助您组合适当的原料。

需要的工具

本书介绍的测试管理方法主要涉及到 5 种基本工具:

- ◆ **完整的测试计划** 详细的测试计划是一个水晶球,它使您能够预见和预防可能

产生的危机。这种计划涉及范围、质量风险管理、人员、资源、硬件逻辑、配置管理、时间安排、测试阶段、主要里程碑和阶段转换,以及预算问题等。

- ◆ **设计良好的测试系统** 一个好的测试系统可以(很有效地)发现可能破坏市场产品的错误。它还具有内外一致性,易于学习和使用,并建立在一组性能好而兼容的工具之上。我使用“良好的测试系统结构”来说明这类系统的特征。“结构”一词可以培养测试小组内部对测试开发的一种全局的结构化观点。它也向管理者传达了创建一个良好的测试系统将涉及具有艺术性和永久性的完美结构的开发。
- ◆ **基于状态的错误跟踪数据库** 在测试过程中,您和测试小组将发现许多故障(也称为“问题”)、缺陷、错误、问题、过错,以及其他不太容易描述的问题。试图在脑海中或者单个文档中记住所有这些错误可能会招致灾难,因为测试人员之间不能有效地进行交流,或者与开发小组进行交流,从而无法帮助提高产品质量。您需要一种方法来跟踪每个错误的系列状态,直到完全解决这个问题。我还将向您展示如何建立和使用符合一个有效而简单的数据库来达到该目的。这个数据库还可以概括信息丰富的图表中的错误,该图表可以告诉管理部门有关项目的测试完成、产品稳定性、系统周转时间、有问题的子系统,以及根本原因的信息。
- ◆ **综合测试跟踪电子表格** 除了跟踪错误外,您还需要跟踪每个测试实例的状态。操作系统会在使用特定的硬件时崩溃吗?以某种格式保存文件花费的时间太长吗?软件或硬件的哪一个版本未通过重要的测试?一个电子表格的一组简单的工作表就可以跟踪每一个测试实例的结果,向您提供回答这些问题所需的细节。详细的工作表还可以积累为摘要工作表,向您显示项目的整体情况。通过的测试实例的百分比是多少?阻塞了多少测试实例?测试集实际需要多长时间来运行?
- ◆ **简单的变化管理数据库** 您是不是常常询问:“我们的时间表怎么会受到如此沉重的打击?”诸如硬件或软件交付日期的偏差、阻塞测试实例的特性缺乏、不可用的测试资源,以及其他“微小”的变化之类的偏差都可能造成伤害。当测试过程延迟时,整个项目就会延迟。虽然无法防止测试延迟事件,但是您可以跟踪它们,这将使管理部门能够尽早注意到延迟,并有效地解决问题。本书提供了一个简单有效的数据库,它能够使危机时刻不会成为您的下一场噩梦。

通过使用 Microsoft Excel、Microsoft Word 和 Microsoft Access 之类基于 Microsoft Windows 的应用程序,我实现了这些基本的工具。我已经使用它们在旅馆房间和

世界各地的飞机场休息室里通过便携式计算机同时管理了多个项目。这些工具简单而有效,它们符合行业标准,并使您与权威软件和硬件供应商的最佳测试管理实践和工具保持一致。我使用这些工具来组织自己对项目的看法,开发有效的测试计划和测试集,执行动态高科技开发环境中的计划,跟踪、分析结果,以及向项目管理者展示结果等。

需要的资源

为了与我们的烹饪比喻保持一致,您还需要一些原料(即资源)以成功地烹饪出菜肴。在这个测试“菜谱”中,我将向您展示如何组合执行测试项目所需的资源。这些资源包括以下部分或全部项目:

- ◆ **实用的测试实验室** 好的测试实验室为人们(以及计算机)提供舒适安全的工作场所。这个实验室远非复活节后第一个星期日的藏匿处,它需要多种方式来与开发小组、管理部门以及其他人员交流。您必须确保它具有足够的软件和硬件,以让测试人员有效地工作,您还必须保持软件和硬件更新为适当的版本级别。请记住,它是一个“测试”室,您需要让工程师能够很容易地跟踪有关系统配置的关键信息。
- ◆ **测试工程师和技术人员** 您需要一个由勤奋工作的合格人员组成的小组,按项目或(和)技能来组织。寻求好的测试工程师可能比寻求好的开发工程师更困难。如何区分初出茅庐的测试天才和将使您这位管理者生活于冲突、危机和效率损失的地狱之中的特殊人物?有时,这二者之间的界限可能比您期望的更微妙。一旦您建立好测试小组,工作就真正开始了。怎样激励小组成员以很好地完成工作?如何拆除可能摧毁动力的地雷?
- ◆ **签约人和顾问** 作为测试管理者,您可能会使用“外围人员”,聘请计时工作人员。我将帮助您给普通的高科技临时人员分类,理解怎样促进他们的工作,并解决围绕他们的情绪问题。何时需要签约人?签约人关心什么?应该设法“留”下优秀的签约人吗?怎样知道自己什么时候需要顾问?
- ◆ **外部测试实验室和销售商** 有些情况下,在自己的测试实验室之外做一些测试是很有意义的——例如,当您需要处理测试大量工作量或疑难问题时。您可能还可以通过利用外部资源提供的技能、基础结构和设备来节省时间和资金。但是,这并非微不足道。这些实验室和供应商能为您做些什么呢?如何才能利用他们来减小测试项目的规模而不会导致危险的遗漏呢?如何将他们的过程和结果在您自己的过程和结果上有所反映呢?

当然,在可以使用其中任何资源之前,您必须将他们集合起来。正如您可能已经学习到的,管理者对所需的工作设备和测试材料上的花费从不吝啬。请记住,我还包括了一些建议来帮助您获得确实需要的资源的批准。

使用本书

本书中没有什么是在建立在“科学真理”、盲目的学习、学术研究,甚至智慧的灵感基础之上的。它只不过是一些已经在我管理的几十个测试项目中发挥作用(并将继续发挥作用)的知识。您可能会“照搬”这些方法,也可能会有所选择的应用。您可能会发现所有(或部分)方法是非常有用的。

同样,需要记住的是,这不是一本有关测试技术、测试理论或者开发过程最新发展水平的书。相反,这是一本根据实践编写的有关测试管理(硬件和软件)方面的书。就开发过程(“最佳实践”或公司的实践)来说,我所做的唯一假设就是,作为测试管理者,您被卷入到一个具有足够的交付时间来做必要的测试开发的开发项目。虽然编写一本有关管理测试流程的书而不定义该流程的每一个方面,看起来有些奇怪,但是,我认为详细的定义将限制本书的实用性。这里介绍的工具和提示的适用范围从特殊的、无序的和随意的操作,到成功的、有序的、经过良好实践的商业硬件和软件公司。您将发现这些工具很容易应用于方法友好的环境中,但是,在许多情况下,当您被迫实现这些工具以强行整理混乱的形势时,这些工具可能最大限度地发挥其作用。

当然,在一定程度上,我不可能只讨论测试管理而不讨论测试技术。因为硬件和软件测试技术不同,所以您可能会发现我所提供的术语表达的不是很清楚,或者我所介绍的用途您并不熟悉。我包括了一个词汇表来帮助软件测试者理解硬件示例,反之亦然。此外,因为测试管理者常常既是技术领导又是管理者,所以应该确信您理解并使用了用于特定测试类型的最佳实践,尤其是在测试技术方面,这一点是很重要的。

本书取材于我的经历,包括好的经历和糟糕的经历。糟糕的经历(我很少使用)旨在帮助您避免我曾犯过的一些错误。我力图使讨论变得轻松一些,因此包括了一些趣闻轶事;附录中列出了我所应用的理论背景及其出处。

我发现自己通过示例学习的效果最好,所以我在本书中包括了许多示例。因为书中介绍的工具适用于硬件和软件测试,所以我将许多示例都建立在下面这两个假想项目之一的基础之上:

- ◆ 大多数软件示例都涉及到开发一个基于 Java 的字处理软件包 SpeedyWriter(由 Software Cafeteria, Inc. 公司编写)。SpeedyWriter 拥有功能齐全的字处理程序的所

有常用功能,以及网络文件锁定、Web 集成和公钥加密等功能。SpeedyWriter 还包括其他供应商的各种 JavaBeans。

- ◆ 大多数硬件示例都引用一个名为“DataRocket”的服务器的开发,由 Winged Bytes, LLP. 开发。DataRocket 提供了功能强大、性能较高的文件和应用程序服务,以及作为 LAN 客户的 Web 主机。它在未知的 CPU 上运行多个操作系统。与第三方软件一起,Winged Bytes 计划集成美国制造的 LAN 卡和台湾的 SCSI 控制器。

至于本书中讨论的工具,本书附带光盘中包含了一些模板,您可以用它来实现自己的工具。随着您阅读各个章节,您可能想要打开并查看光盘中相应的模板,以便深入地理解这些工具的工作方式。阅读完本书,您将会发现光盘中的模板作为单独的对象是很有用的。

最后,万一您尚未发现这一点,那么测试不是资源和时间的泛滥场所。我发现,将测试重点放在项目管理者真正重视的方面是很关键的。过去,我常常以被各种事件弄得手忙脚乱而告终,花费大量的时间只处理了一些琐碎的小事,而重要的事情却逃出了我的注意范围。这些经历告诉我,要识别和注意少数重要的方面,而忽略多数次要的方面。这里展示的工具和技术可以帮助您做到这一点。大约 1/4 的计算机测试组织在他们成立最初的两年内就解散了;而本书将帮助您成为剩余的 3/4 中的一员。

虽然这显然不只是为了保住工作,但成功的测试管理对不同的人会有不同的意义。在日常工作中,我衡量成功的好处的依据是心灵的平静、压力的减轻,以及由实际管理测试领域所带来的职业形象的提高,而不是忙于应付特殊环境中发生的无穷无尽的危机。我希望这些工具和思想将有助于您成为一位成功的计算机测试专家。

目录

致谢	1
简介	3
第一章 定义板块内容：测试系统的基础	1
1.1 想要测试什么：扩展的测试工作	2
1.1.1 从显微镜到望远镜：测试粒度	2
1.1.2 快跑还是慢行：测试阶段	4
1.1.3 第一次划分	9
1.2 应该测试什么：考虑质量	9
1.2.1 三个盲人和一头大象：您能够为质量下定义吗？	10
1.2.2 不同质量体验的危害	11
1.2.3 经典和超越：非正式的质量风险评估方法	13
1.2.4 失败模式和效果分析：理解质量风险的一种正规方法	24
1.3 能够测试什么：计划、资源和预算	28
1.3.1 强行灌输：使测试计划适合于项目	29
1.3.2 估计资源并创建预算	31
1.3.3 洽谈一个可行的测试项目	33
第二章 策划和展示计划：测试计划	35
2.1 为什么要撰写测试计划？	35
2.2 测试计划的数量有多少？	36
2.3 使用草稿来激发讨论	37
2.4 测试计划模板	37
2.4.1 概要	37
2.4.2 领域	38
2.4.3 质量风险	39

2.4.4	提议的重大事件时间表	40
2.4.5	转换	40
2.4.6	测试配置和环境	42
2.4.7	测试执行	44
2.4.8	风险和意外事故	46
2.4.9	更改历史	47
2.4.10	参考文档	47
2.5	提交计划	47
2.6	适可而止：一个警告	48
第三章	测试系统的结构、方案和范围	49
3.1	测试系统结构和工程	49
3.1.1	“操作”组件：定义	50
3.1.2	它不是圣保罗大教堂，但是	52
3.1.3	不是一个美术品：测试系统的性能	53
3.1.4	测试系统不是孤岛：测试人员与测试系统	56
3.2	系统的砖块和灰泥：测试方案模板	57
3.2.1	集中起来：测试方案页眉	58
3.2.2	测试步骤	59
3.2.3	小结：测试结果摘要	60
3.2.4	DataRocket 的一个测试方案示例	61
3.2.5	要多详细？权衡模糊性	63
3.3	避免可怕的“测试逃逸”：范围和回归测试间隙	64
3.3.1	具有最好意图的糟糕的范围决定	65
3.3.2	您此刻在测试开发部门正建立的区域吗？	66
3.3.3	将质量风险与测试方案联系起来	66
3.3.4	配置范围	68
3.3.5	错误范围	69
3.3.6	回归测试间隙	71
3.4	“这里有一个需要学习的教训”：测试方案累积改进	81
3.4.1	对失败的反应	83
3.4.2	采用“最佳实践”	83
3.4.3	使用特别测试	84
3.5	无法进行所有测试：确定不进行哪些测试	84
第四章	昆虫学中令人激动的职业：错误跟踪数据库	87
4.1	为何如此麻烦？一个正式的错误跟踪系统实例	88

4.2 哪些似乎是问题？失败描述	89
4.2.1 更像海明威而不像福克纳	90
4.2.2 在沙地上划一条线：隔离与调试之间的区别	93
4.3 灵活的报告：开始组建数据库	94
4.4 致命的少数和无足轻重的多数：划分重要性等级	97
4.4.1 严重性	97
4.4.2 优先级	97
4.4.3 抓住危险优先级中的重要性	98
4.4.4 向数据库中添加重要性度量	98
4.5 在错误跟踪中放入跟踪：添加动态信息	100
4.5.1 使用“阶段”来管理错误的生命期	100
4.5.2 强调所有权和可计算性	101
4.5.3 不知道太多(错误)历史？跟踪状态更改	102
4.5.4 在合适的位置放入动态字段	102
4.6 完成联系：专注于错误	104
4.6.1 Subsystem	104
4.6.2 Configuration	105
4.6.3 Close Date 和 Resolution	106
4.6.4 Root Cause	106
4.6.5 添加错误焦点字段	109
4.7 从错误跟踪数据库中找出规律	111
4.7.1 清除缺陷的进展如何：打开/关闭图表	112
4.7.2 错误为何发生：根本原因图表	116
4.7.3 开发人员如何响应：关闭周期图表	117
4.7.4 什么中断了：子系统图表	119
4.7.5 一个关于规律和图表的注释	120
4.8 管理错误跟踪	121
4.8.1 策略及错误数据的滥用	121
4.8.2 困境	123
第五章 管理测试方案：测试跟踪电子表格	127
5.1 创建一个要求最简单的测试跟踪电子表格	128
5.2 增强功能	130
5.2.1 标识测试套件和实例	130
5.2.2 添加日期和时间信息：计划与现实	131
5.2.3 收集用于可靠性计算的数据	131
5.2.4 提高测试方案状态的精确性	132

5.2.5	计算加权的失败	133
5.2.6	记录所有权	134
5.2.7	运行重要的测试方案	135
5.2.8	添加其他摘要	135
5.2.9	组合数据	135
5.3	添加测试方案细节	136
5.4	启动测试跟踪系统	137
5.4.1	小问题	137
5.4.2	大问题	139
5.4.3	没有问题!	141
第六章	决定性时刻的提示和工具：管理动态情况	143
6.1	勿为琐事忧心：居于万事之上	143
6.1.1	获得所有事实后继续前行：确定的渴望、进度的需要	143
6.1.2	相关性、计划和提醒：跟上进度的重要性	144
6.1.3	它不会交付自身：修订与发布过程	145
6.1.4	它也不会安装自身：配置测试环境	145
6.1.5	“小脑怪”是您的朋友：审核和更新测试结果	146
6.1.6	避开陷阱：不要陷于调试	147
6.1.7	当测试失败时：将类型 I 和类型 II 错误减到最少	148
6.1.8	愉快的龙舟节：在决定性时刻、节假日和文化冲撞之际	150
6.2	蜘蛛的连接网：管理测试硬件和软件配置逻辑	152
6.2.1	各个部分及其连接方式：实体关系图	152
6.2.2	从图表到架构：实现逻辑数据库	155
6.2.3	预算和计划：提早使用逻辑数据库	156
6.2.4	跟踪软件配置	166
6.3	期望意外情况：变化管理数据库	169
6.3.1	该怎么办呢？使用(和滥用)变化管理数据	170
6.3.2	简单就好：变化管理数据库	170
第七章	充实和管理测试实验室	173
7.1	需要测试实验室吗？	174
7.2	选择和计划实验室区域	175
7.3	测试实验室清单	179
7.3.1	一个示例目录清单	179
7.3.2	进一步考虑补充实验室	183

7.4 安全与跟踪考虑	184
7.5 管理设备和配置	185
7.6 保持测试环境的整洁	187
7.7 人的因素	188
7.7.1 安全的实验室就是有效率的实验室	189
7.7.2 对实验设备的损坏	190
7.7.3 实验室效率	191
第八章 组织和管理测试小组	193
8.1 决定测试小组：需要多少人做什么工作？	193
8.1.1 规模	194
8.1.2 技能	196
8.1.3 位置	197
8.2 专家或项目资源？组织模型	198
8.3 工作的正确人选：测试工程师的条件	201
8.3.1 测试专家与内容专家	202
8.3.2 职业的悲观主义	203
8.3.3 平衡的好奇心	204
8.3.4 没有隔挡拉皮条的人	205
8.4 工作中的错误人选：不适合测试工程师的品质	206
8.4.1 魅力探求者	206
8.4.2 懒惰	206
8.4.3 卡通片里的胆小鬼	207
8.5 提出指责：激励测试小组	208
8.5.1 不要让别人将您的小组作为一个擦鞋垫来使用	208
8.5.2 注意极限问题	210
8.6 特洛伊木马和其他错误：如何挫伤测试人员的积极性	213
8.7 扩展您的才能：使用临时的工具和专家	214
8.7.1 评估临时工作人员的角色	214
8.7.2 使用长期临时人员	216
8.7.3 聘请签约人	219
8.7.4 引进专家	222
第九章 政治的胜利：对测试管理者的组织挑战	225
9.1 堂吉珂德,质量冠军：总之,您的工作目标是什么？	225
9.2 您适合的位置：组织中的测试小组	228

9.3 其他什么工作适合? 向测试增加其他功能	230
9.4 与其他管理者合作: 测试管理的两个其他方面	232
9.4.1 管理上层	233
9.4.2 管理外部	242
9.5 在黑暗中测试: 在没有文档的情况下您应该如何继续?	245
9.6 测试并非孤岛: 外部环境对效率的影响	247
9.6.1 油门	247
9.6.2 刹车	249
9.7 解雇通知书: 临时解雇和停业清理	252
9.8 显示结果: 正确的消息, 正确交付	253
9.8.1 提交坏消息的好方法	254
9.8.2 准确和听众的重要性	254
9.9 “您可以告诉先驱者……”在测试上过早采用的效果	256
第十章 包括其他人: 分配测试项目	261
10.1 选择伙伴	262
10.1.1 供应商	263
10.1.2 第三方测试公司	265
10.1.3 销售办事处	267
10.2 计划分布式测试工作	268
10.2.1 评估能力	268
10.2.2 理解花费	270
10.2.3 整理、协调和划分测试项目	271
10.2.4 组织逻辑	272
10.2.5 处理映射问题	273
10.3 管理分布式测试工作	275
10.3.1 监视测试执行	276
10.3.2 交流状态和更改指导	276
10.3.3 处理政治问题	277
10.3.4 小心文化冲突	278
10.3.5 建立和维持信任	280
附录	283
词汇	287