

TURING

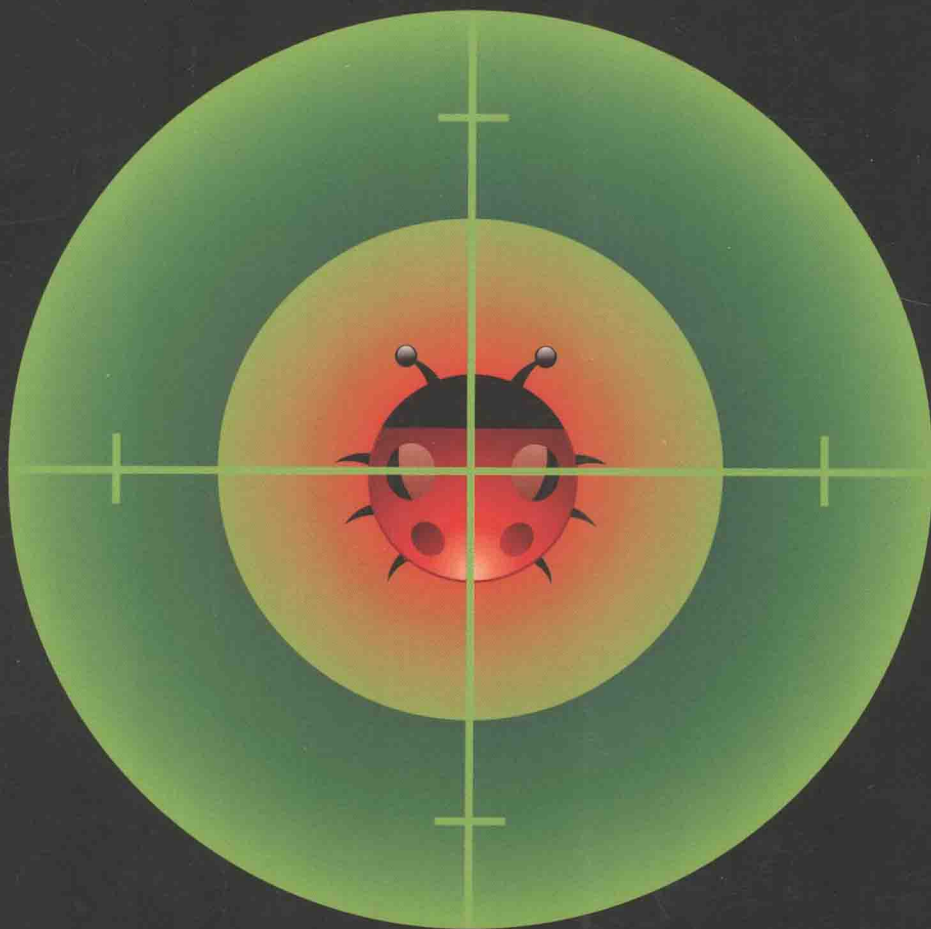
图灵原创

微软一线测试专家实战精华
全面涵盖软件测试实用技术

软件测试实战

微软技术专家经验总结

史亮 ◎ 编著



人民邮电出版社
POSTS & TELECOM PRESS

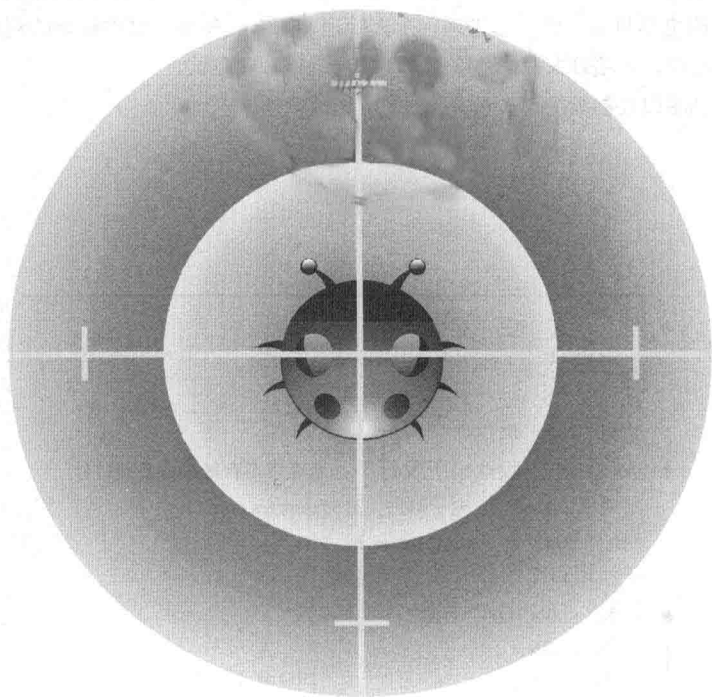
TURING

图灵原创

软件测试实战

微软技术专家经验总结

史亮◎编著



人民邮电出版社
北京

图书在版编目 (C I P) 数据

软件测试实战：微软技术专家经验总结 / 史亮编著.

— 北京：人民邮电出版社，2014.4

(图灵原创)

ISBN 978-7-115-34584-4

I. ①软… II. ①史… III. ①软件—测试 IV.

①TP311.5

中国版本图书馆CIP数据核字(2014)第032063号

内 容 提 要

本书从多个角度讨论了测试人员的实际工作，包括缺陷报告、测试文档、测试建模、测试设计、测试自动化、研究产品、研究项目环境、测试管理、个人管理、实践案例等。书中崭新的观念与技术将有助于读者更好地提交缺陷报告，在项目末期的缺陷压力下更好地做回归测试。

本书适用于测试新手以及初级测试人员。

◆ 编 著 史 亮

责任编辑 王军花

执行编辑 张 霞

责任印制 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>

三河市海波印务有限公司印刷

◆ 开本：800×1000 1/16

印张：22.25

字数：497千字

印数：1-4 000册

2014年4月第1版

2014年4月河北第1次印刷

定价：69.00元

读者服务热线：(010)51095186转600 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京崇工商广字第 0021 号



推荐序

Recommendation 1



2014年年初，正值学期结束之际，工作比较忙，就利用这次从银川飞往上海的途中，完成了一项非常有意义却很有挑战的任务：为微软总部资深测试工程师史亮博士所著的《软件测试实战》写序。飞机上没有网络、电话，倒是一个很封闭、不受干扰的session，可以集中注意力做好一件事；从另一个方面看，也说明自己对写此序非常重视，希望不愧对朋友所托，能够较好地完成这一“艰巨”的任务。

差不多两年前，我也曾为作者和高翔所著的《探索式测试实践之路》一书写过序，但这本新书有很大不同，涉猎之广几乎覆盖了测试工作各个方面，从缺陷报告、测试建模、测试需求分析与设计到团队建设、个人管理等。从单一测试方式的测试实践走向全面的软件测试实践之路，这期间需要更多的积累，需要更高、更广的视野，由此体现出作者全面的测试知识、独到的思考和丰富的经验。本书不仅内容丰富，入木三分，而且对读者的指导体贴入微，循序渐进地给予具体的示范性操作。例如，通过一周的测试过程指导测试人员如何做好具体的测试工作。本书侧重测试实践，如在讨论“面向调试的测试代码”、“系统测试的测试开发”时，把作者自己实践过的经验一条一条地摆出来展开讨论，并通过列出测试人员容易碰到的问题，逐条解答，以帮助读者解决一些复杂的测试难题，如根据语境（上下文）如何完善组合测试的模型、启发式测试Oracle应用等。本书还介绍了一些新的测试概念或方法，如大规模自动化测试（HiVAT），系统生态图的应用，风险管理中的长回路、短回路等。除此之外，我们还能从本书学习到大量测试想法、清晰的测试逻辑思维、各种测试方法等，极大地增强我们的测试技能。

近几年，作者在美国微软公司总部工作，本书分享的思想、方法和经验多数来自于软件业最发达的欧美，吸收了不少欧美测试大师的卓越思想和优秀实践，引用了近百项资料文献，所有文献都是英文的，时间跨度也很长，从1972年到2012年，长达40多年。可以这样说，本书为我们省去了阅读国外文献的大量时间，仅从这一点来看，这本书也极具价值，更何况本书对一批测试大师的智慧进行了提炼，并应用到实际的案例中。这些测试大师包括我们熟知的James Bach、

Jonathan Bach、Cem Kaner、Michael Bolton、James Whittaker、Elisabeth Hendrickson、Rikard Edgren、Harry Robinson，等等。因此，本书籍此分享了他们贡献给测试知识宝库的一些精华内容，如：

- 产品研究的7大元素 SFDAPOT；
- 启发式测试计划语境（上下文）模型HTPCM；
- 启发式测试策略模型（HTSM）；
- 探索式测试管理方法SBTM（Session-Based Test Management）；
- 测试技术分类体系；
- 敏捷四象限、测试自动化金字塔；
- 基于缺陷模型的快速测试方法；
- 基于部署图的测试指导词；
- 测试Oracle指导词FEW HICCUPPS。

软件测试不同于硬件测试，不仅在于软件的复杂性、软件需求的善变性，而且在于软件的社会性，在一个软件中融入了更多的社会因素，如不同用户的软件应用体验、受业务环境影响的业务操作、受人们心理影响的业务逻辑变化。基于这样的背景，基于上下文驱动测试（Context-driven test）的方法体系引起人们足够的重视，我也深感作者受这个体系（以James Bach、Cem Kaner、Michael Bolton等为代表的流派）的影响比较大，包括对软件测试的理解，更倾向于这样的观点：测试是一种服务测试人员通过服务团队来体现自己的价值；测试是一种技术调查，其目的是向干系人提供有关产品质量的实验消息。本书依旧继承了《探索式测试实践之路》的测试思想，在测试技术这一章，并没有讨论功能测试和性能测试的传统方法和技术，而更多是从“上下文驱动学派”的启发式测试策略、快速测试、探索式测试等方式方法出发，来介绍测试技术，包括情景测试、漫游测试、肥皂剧测试等技术，使测试更有趣，更能适应软件测试的社会性。

本书从缺陷报告开始讨论，虽然编排不同于一般的测试书籍，但也合乎逻辑，缺陷发现和报告是一个测试人的基本能力。基于时间和方法技术的限制，一般应用系统的测试还无法做到百分之百的测试，所以迫使人们常常从反向思维角度思考测试，就是尽可能发现缺陷，缺陷发现越多且能被修复，产品质量就越高。从这一思维方式出发，发现缺陷是软件测试的基本目标之一，或者说，发现缺陷的测试是成功的测试。作者在书中分享了自己的真知灼见，如高质量的缺陷报告来自于高质量的测试。

当敏捷开发席卷世界时，人们对文档重视程度越来越低的时候，本书把测试文档放在很重要的地位，体现了作者的独立思考和务实的态度，不随波逐流，从测试自身要求出发，强调“测试文档是持续演化的工具”，把探索式测试的思想“测试设计、执行、评估（分析）、学习”应用到文档的改进和维护中，并通过从测试计划、设计规约、检查列表到各种形形色色文档的介绍，也让读者更全面、更深刻地理解软件测试。其次，已有测试书籍对测试建模及其应用介绍都较少，但本书不仅在第4章用一章的篇幅来介绍各种测试建模方法及其应用，而且在第3章、第7章等都有所涉及，揭开了测试建模的神秘面纱，让测试人员不再拒测试建模于千里之外，促进测试人员

在日常测试工作中能够习惯地运用测试建模来解决复杂系统的测试问题，不断地改进测试。

自动化测试永远是软件测试的核心主题之一，本书也不例外，谈到自动化测试策略和实践，如用了较多篇幅讨论面向调试的测试代码、系统级别（包括系统GUI测试）的自动化测试开发。但是，作者用了更大篇幅来研究“产品”和“项目”，因为这些的确是测试的基础，产品是测试的对象，不能真正理解产品、不熟悉产品功能特性和技术实现，测试是不可能做好的；而每个要执行的测试任务都是为了达到测试目标，而测试目标及其实现都直接和其项目的环境因素相关联。

我在阅读本书时，不仅产生了上述这么多的共鸣，而且还有更多的收获和体会。例如，在阅读8.2节内容时，就勾起了我在webex工作的某些回忆。当初我们使用自己开发的项目和缺陷管理系统QAForum，这个系统有一个很大的优点，就是用户可以写SQL语句，构造自己想要的任何形式的、灵活抽取不同数据的缺陷报告。

最后，希望读者在阅读本书时能有更多的思考，并根据本书的指引，阅读相关的资料，进一步地思考、实践、再思考、再实践，一定会受益匪浅。

朱少民

同济大学软件学院教授

推荐序

Recommendation 2



非常荣幸受邀为史亮的这本新书作序。毫无疑问，史亮是一位优秀的思考者，从本书信手拈来的各种案例中就能够清楚地看到这一点：从离开某项目后其他人接手的困难中，史亮思考和总结出了测试工程师工作的依赖性；从代码覆盖的案例中引申出覆盖率的作用和陷阱……能够在国内的测试领域看到这样一本真正以个人思考为主的原创性图书，实属不易。我相信史亮非常好地贯彻了他在书中“且行且思”这一节中提到的理念：一方面，它暗示软件人员的职业生涯是“漫漫长路”，需要坚持不懈地跋涉；另一方面，它指出持续的努力应该伴随持续的思考，执行与反思缺一不可。

我从1998年偶然进入软件测试领域，在其中投入了十多年的精力，对软件测试的不少方向都有所涉及，期间经历过多种类型的项目，和不同的人员合作过，尝试过各种测试技术和方法。虽然从2011年起软件测试不再是我的主要方向，但我有幸经历了中国的软件测试行业迅速发展的十年，对于软件测试本身有着自己的一些理解和思考。

当然，即使在这个行业投入了十多年的时间，我仍然无法为“该怎么做软件测试”这个问题给出简单明确的回答。从2011年开始，我的主要角色变成了互联网组织的整个研发团队的管理者，这个位置使得我可以不再站在一个纯粹测试者的角度看待问题。站在组织的角度看待测试，最关心的问题不再是“该怎么做软件测试”，而是“软件测试该如何促进组织”。以促进组织（生产率等）为目标的测试，自然不仅仅需要测试工程师的工作，还需要开发工程师、产品经理等各种角色共同参与进来。有代表性的互联网企业（如Google、Facebook）不约而同地采用了小规模测试团队（甚至不设置单独的测试团队），越来越多的开发工程师开始关注测试和强调测试技能（包括对可测试性的关注）。在这样的趋势之下，现有测试行业的主要从业者——测试工程师自然会面临压力和机遇。压力不言而喻：来自组织和开发者的更高期待使得测试工程师不得不具有更全面的视野、更扎实的技术能力，以及更有价值的产出；而机遇，则是测试工程师可以有机会摆脱一直以来圈住自己的“发现缺陷”的定位，真真正正地为组织创造更大的价值。

然而，“创造真正的价值”需要测试工程师系统的能力，仅靠亦步亦趋地跟随某个流程无法达成；幻想通过盖世神兵（某个工具）或是幻想掉下山崖拿到神秘的内功心法也无法达成。有人说测试工程师是个“越老越吃香”的行业，我却要说，这个行业需要的不是资历，而是“经验”。即使你把手头上的工作重复十年，没有思考，不经过系统的学习和提炼，你的经历永远只能是低层次的重复而已。该如何学习？该如何思考？该如何系统化你的经验？

不得不说，就我看到的状况而言，大部分测试工程师还并未理解这个巨大的转变。仍然有不少人认为测试就是根据流程完成任务，或是仍然执着于测试与开发的界限，画地为牢地守着那些可怜巴巴的“测试技术”。从这个意义上说，史亮这本书适逢其时，它不是一本以说教口吻教你“该如何一步一步做好测试”的书，也不是一本告诉你测试领域又出现了多少新名词的书，它用作者本人的经验，和你分享一个在测试行业勤于思考的人对测试的理解。

这本书不是一本让你用来模仿的书，而是一本让你有所思考的书。史亮在本书中和盘托出的，不是自己的武功招式，而是自己的心法。如果想要从本书中得到最大的价值，我建议你静下心来，仔细品读史亮在本书中娓娓道来的思考，在他的带领下，和他一起领略思考的乐趣吧。

段念

豆瓣工程副总裁

2014年2月16日于北京

推荐序

Recommendation 3



我和史亮从2009年就认识了，对于史亮在测试技术上的钻研精神我非常敬佩。前两年我有幸和史亮合作一起编写《探索式测试实践之路》一书，在那段测试技术交流的日子里，史亮对于软件测试的前沿测试技术和测试理念的理解让我印象非常深刻。他还积极地在工作中实践最新的测试技术，分享自己的心得体会，一直是我学习的榜样。

通过查看这本书的参考文献就知道，史亮很认真地编写这本关于软件测试技术的实战书籍，我也仔细地阅读了该书相关章节，最大的感受就是这本书的可操作性比较强，与传统测试书籍的理论知识较多而实际案例较少相比，这本书更能让我们体会到理论和实践的相互结合和相辅相成的关系。史亮在微软美国总部接触到了这么多前沿的测试技术和理论并将它们付诸实践，这些宝贵的经验都沉甸甸地包含在这本书里，特别是第4章的建模技术和第5章的测试分析技术，史亮将业界大部分测试技术和分析方法通过简单实用的方式描述出来，方便读者学习和应用。我认为，读者看了这本《软件测试实战》后，80%的软件测试书籍和测试文章都可以不用再去阅读了，因为精华的内容和微软一线测试专家亲自实践的心得都涵盖在这本书里。

史亮不仅在测试开发和测试管理上有自己的独特见解，也善于在工作过程中思考如何更好地做软件测试，如何更好地分析产品、提高测试效率。我和史亮都认为，软件测试分析、设计、管理在整个项目周期中是个动态的过程，并且存在较好的手段来提高和控制这个动态过程，《软件测试实战》将告诉读者如何更好地控制和提升整个动态过程。同时这本书也传递了这样一种理念：软件测试不仅仅是测试工程师的个人舞台，更需要和项目团队、被测项目/产品、个人管理充分融合和滋润，才能完美地体现软件测试的价值。

目前国内的很多大公司都有测试开发工程师这个岗位，但是很多人会认为测试开发工程师更多的是从事开发的工作，而不是测试的工作。史亮将自己在微软总部的亲身实践经历告诉读者，

如何来理解测试和开发的关系，如何把测试开发工作效益最大化，如何做有效的自动化测试，而不是盲目的自动化测试。

最后，我极力推荐这本《软件测试实战》，因为它不仅会让你学习到最新的测试技术和实践成果，而且会让你体会和理解一个在测试技术上不断突破自我的工程师的心路历程。

高翔

淘宝资深测试工程师，《探索式测试实践之路》合著者

前 FORWORD

我在攻读博士学位时开始研究软件测试，毕业后一直任职测试工程师，从事第一线的测试工作，不知不觉已有十余年的光阴。在此期间，我阅读过大量的测试文献，参与过多个不同类型的测试项目，既学到了许多有价值的方法，也观察到了一些不太有效的实践，既通过努力获得过成功的经验，也从错误中得到了宝贵的教训。随着经历的增长，我渐渐构建出自己的知识体系，从实践中打磨出一批指导测试工作的策略与经验。自然而然，一个朴素的想法浮现出来：如果将我的所学所知分享给更多的测试工程师，想必能帮助他们节省学习与积累的时间，以更快地提高测试水平。在该想法的驱动下，我广泛地阅读了测试文献，深入地反思了自己的实践，并开始了漫长的写作。成果就是您所读到的这本书。

本书分享了我从事软件测试工作所学到的知识和所总结的经验，旨在帮助测试人员建立正确的观念，并掌握一批切合实战的测试技术。一方面，我总结了测试专家的理解和方法，将其精华内容综述在本书之中，以帮助读者提高学习效率、快速地掌握综合性的技能。另一方面，我努力将自己的经验和反思融入书稿，使它反映出我在工作中使用的策略、方法和技巧。总之，这是一本注重实效的书，尝试用理论结合实践的方式来解决现实的问题。

本书的组织方式

在概述一些基本测试观点（第1章）之后，本书按照“启发式测试策略模型”（参见4.2.1节）的基本元素来展开论述：观察到的质量（第2章）、测试设计（第3、4、5、6章）、产品元素（第7章）和项目环境（第8章）。随后，讨论了测试人员如何参与团队工作（第9章）并实施个人管理（第10章）。

- 第1章阐述我对软件测试的基本观点，介绍了我的测试价值观。作为指导原则，它们将贯穿全书的内容。

- 第2章讨论了测试人员最主要的工作产出——缺陷报告，介绍了一批实践方法，帮助测试人员高质量地报告缺陷，并利用该过程来改进测试设计。
- 第3章讨论作为测试辅助工具的测试文档，介绍了一些编写和维护文档的原则和方法。通过分析一批具体的测试文档，展示出测试设计的迭代性和多样性。
- 第4章介绍指导测试设计的模型，通过应用组合测试，阐述了如何根据项目语境来完善测试模型，然后介绍了测试建模的基本原则和常用方法。
- 第5章介绍一个测试技术分类系统，以概览各类测试技术。然后，讨论了一批有价值的测试技术，包括启发式测试先知、漫游测试、快速测试、情景测试等。在此基础上，强调了测试人员应该多样地选择测试技术，以动态地优化测试的价值。
- 第6章讨论了测试开发的基本分类，然后针对自动化测试、计算机辅助测试和大规模自动化测试，阐述了它们的基本概念、设计目标、开发策略和实作方法。
- 第7章讨论如何从测试视角来研究软件产品和业务领域，介绍了静态分析、动态分析、关系人研究、需求评审、测试调查、网络调研、领域研究等研究方法。
- 第8章讨论如何从测试视角来研究项目环境，介绍了团队分析、缺陷分析、源码分析、构建分析、自动化测试分析、基于风险的测试等研究方法。
- 第9章探讨测试人员如何有效地在团队中工作，以及如何恰当地实施测试管理。针对一些常见任务，如小组协作、测试计划、工作量估算、软件度量等，提出了一批注重实效的方法。
- 第10章分享了我实施个人管理的基本方法，包含时间管理、个人学习、经验积累、专业发展等。

目标读者

本书的主要读者是具备一定测试经验、想要进一步提高测试能力的测试工程师。全书综合了测试行家的专业建议和我的实践经验，探讨了测试价值观、测试设计、产品研究、项目研究、团队协作、个人管理等多个方面，能够帮助读者更好地理解软件测试，并提高实践水平。本书的内容面向广大的测试社区，并不要求读者掌握特定的背景知识。我希望它能够适合大多数测试人员，并通过解决现实问题来引起读者的共鸣。

此外，本书是很好的软件工程课程、软件测试课程和测试培训的参考资料，能帮助本科生、研究生和测试学员更好地理解真实的软件测试。一些测试人员在工作后会发现，课程教材和培训教程所描述的内容并不切合实际的测试工作。例如，课本中的测试设计和执行是线性实施的，真实测试流程却是迭代展开的；缺陷报告常常是测试工作的核心产出，教材却没有讨论如何有效地管理缺陷报告；测试人员被要求撰写文档，其成果又被束之高阁，却少有教程讨论如何编写有价值的文档。这时，他们可以参考本书所提出的建议，从而发展出更有效的工作策略。

如何阅读本书

本书第1章讨论软件测试的基本事实和价值观，是全书内容的基础，需要首先阅读。在读完第1章后，读者可以按任意顺序阅读本书。您既可以顺序浏览，以概观软件测试，也可以有选择地阅读感兴趣的章节，并在阅读的过程中参考相关内容。

这是一本关于实践的书，许多建议和方法来自于实践。软件开发专家Ralph E. Johnson指出，“从实践中来的知识在没有实践之前是无法被真正理解的”（practical knowledge has to be experienced to fully understood），测试专家Cem Kaner、James Bach等也认为“你不能掌握测试，除非你重新发明它”（You can't master testing unless you reinvent it）。在阅读过程中，读者需要积极思考本书的方法是否适用于自己的项目，然后将恰当的方法应用于真实的测试，并认真评估其效果。通过练习、评估和反思，读者能够掌握方法的原理和细节，并混入自身经验和其他技术，以演化出新的方法。坚持这样的研究和创新将帮助测试人员走上精通之路。

致谢

作为作者，我诚挚感谢人民邮电出版社和北京图灵文化发展有限公司为我提供创作的平台。感谢本书的编辑王军花、张霞等，你们的专业态度和细致工作提升了本书的品质。

我还要感谢审稿人和推荐人的批评指教，你们的细心评阅和宝贵意见令我受益匪浅。谢谢你们的时间和智慧。

最后，我要感谢妻子汤红红，没有你的理解和支持，我无法完成此书。我还要感谢儿子史泰德用欢笑带给我快乐与能量，你令我想成为一个更好的人。感谢你们让我梦想。

目 录

CONTENTS



第 1 章 软件测试基础	1	2.3.7 注意缺陷报告的可读性	26
1.1 软件的复杂度已经超越了人的理解能力	1	2.3.8 客观中立地书写缺陷报告	26
1.2 软件测试是获取信息的技术调查	4	2.4 对不予修复的缺陷进行上诉	27
1.3 测试是迭代过程	6	2.5 周密地测试缺陷修复	28
1.4 测试人员的工作效率取决于他对软件和项目的理解，而不是他掌握的测试技术	8	2.6 坚持阅读缺陷报告	30
1.5 小结	9	2.7 小结	32
第 2 章 缺陷报告	10	第 3 章 测试文档	33
2.1 报告缺陷是为了让缺陷得到修复	10	3.1 测试文档是持续演化的工具	33
2.2 高质量的缺陷报告来自于高质量的测试	13	3.1.1 测试文档是提供测试信息的一组文档	33
2.2.1 分配测试时间	13	3.1.2 在测试中演化测试文档	35
2.2.2 通过技术调查发现更多的信息	14	3.1.3 注重实效的测试文档	36
2.2.3 处理难以重现的缺陷	17	3.2 形形色色的测试文档	40
2.3 编写高质量的缺陷报告	22	3.2.1 测试计划	40
2.3.1 为每一个缺陷单独提交一份缺陷报告，小缺陷也是如此	22	3.2.2 Google ACC	44
2.3.2 仔细编写缺陷报告的标题	23	3.2.3 测试设计规约	48
2.3.3 像编写详细测试用例那样编写重现步骤	23	3.2.4 功能列表	49
2.3.4 使用缺陷模板来提交缺陷	24	3.2.5 大纲与思维导图	51
2.3.5 在编写缺陷报告时，要考虑缺陷查询	25	3.2.6 表格（矩阵）	53
2.3.6 链接相关的缺陷	26	3.2.7 测试指南	55
		3.2.8 测试想法列表	56
		3.2.9 质量特性列表	57
		3.2.10 操作文档	58
		3.2.11 检查列表	59
		3.2.12 缺陷目录	60

3.2.13 测程表	61	5.6.2 设计用户角色	158
3.2.14 移交文档	67	5.6.3 情景测试与漫游测试	161
3.3 在测试中发展测试文档	67	5.6.4 肥皂刷测试	162
3.3.1 初始测试文档	67	5.6.5 虚拟业务	164
3.3.2 发展测试文档	70	5.7 多样地选择测试技术	165
3.4 小结	71	5.8 小结	166
第 4 章 测试建模	73	第 6 章 测试开发	168
4.1 从组合测试看建模的重要性	73	6.1 测试开发分类	168
4.1.1 组合测试简介	73	6.2 注重实效的自动化测试	169
4.1.2 根据语境来完善组合测试的 模型	76	6.2.1 自动化测试的基本策略	169
4.1.3 测试建模的基本点	83	6.2.2 将测试开发视作软件开发	172
4.2 常用测试建模方法	84	6.2.3 利用自动化测试金字塔来指导 测试开发	175
4.2.1 启发式测试策略模型	84	6.2.4 面向调试的测试代码	177
4.2.2 输入与输出模型	91	6.2.5 系统测试的测试开发	182
4.2.3 系统生态图	96	6.2.6 让自动化测试服务于项目	190
4.2.4 实体关系模型	101	6.3 计算机辅助测试	193
4.2.5 状态机模型	105	6.3.1 “交通工具”的隐喻	193
4.2.6 多种多样的模型	109	6.3.2 选择合适的开发技术	195
4.3 小结	113	6.4 大规模自动化测试	204
第 5 章 测试技术	115	6.4.1 基本概念	204
5.1 测试技术分类系统	115	6.4.2 测试设计	206
5.2 启发式方法	120	6.5 小结	211
5.3 测试先知	122	第 7 章 研究产品	213
5.3.1 测试先知的定义	122	7.1 静态分析	213
5.3.2 FEW HICCUPPS	123	7.1.1 浏览源代码来理解产品实现	213
5.3.3 约束检查	129	7.1.2 分析源代码来帮助测试设计	221
5.4 漫游测试	133	7.1.3 黑盒测试并不是基于无知的 测试	227
5.4.1 基本漫游方法	134	7.2 动态分析	229
5.4.2 基于旅行者隐喻的漫游方法	135	7.2.1 用工具分析产品的行为	229
5.4.3 移动测试漫游方法	139	7.2.2 在调试器中观察软件行为	237
5.4.4 实施漫游测试	142	7.3 业务研究	241
5.5 快速测试	144	7.3.1 理解关系人	241
5.5.1 James Bach 的方法	145	7.3.2 评审需求文档	245
5.5.2 Cem Kaner 的方法	147	7.3.3 通过测试来研究	251
5.5.3 James Whittaker 的方法	149	7.3.4 利用互联网资源	255
5.6 情景测试	154	7.3.5 领域研究	258
5.6.1 基本方法	155		

7.4 研究策略	260	9.3 软件估算	299
7.5 小结	261	9.3.1 测试人员应该估算自己的 任务	300
第 8 章 研究项目	262	9.3.2 用计数和计算作为估算手段	302
8.1 项目团队	262	9.3.3 历史数据是估算的重要参考	303
8.1.1 了解团队组织	262	9.3.4 同时估算最差情况和最好情况	305
8.1.2 语境独立的启发式问题	264	9.4 度量	305
8.1.3 了解团队成员	268	9.4.1 理解度量方法的基本元素	306
8.2 面向测试的项目分析	270	9.4.2 明确度量的目标	307
8.2.1 软件缺陷	270	9.4.3 掌握属性和算法的联系	309
8.2.2 源代码	274	9.4.4 理解度量方法的优点和缺点	311
8.2.3 构建	277	9.4.5 密切关注度量的副作用	312
8.2.4 自动化测试	279	9.4.6 注重实效的计算	314
8.3 基于风险的测试	282	9.5 测试小组	315
8.3.1 通过测试调查风险	282	9.5.1 价值观	315
8.3.2 失败模式	284	9.5.2 团队建设	316
8.3.3 项目级别的风险	285	9.6 小结	318
8.4 小结	287	第 10 章 个人管理	319
第 9 章 团队工作	288	10.1 时间管理	319
9.1 工作风格	288	10.1.1 利用任务清单记录所有工 作项	319
9.1.1 测试人员通过服务团队来体现 自己的价值	288	10.1.2 坚持周计划和每日回顾	322
9.1.2 测试人员应该正直	290	10.1.3 专注是高效工作的前提	322
9.1.3 测试人员的影响力来自于出色 的工作	292	10.1.4 恰到好处的文档化和自动 化	324
9.1.4 信任程序员的努力, 并用技术 调查检验其工作	292	10.2 持续学习	328
9.2 测试管理	293	10.2.1 在工作中学习	328
9.2.1 个人测试计划应该是项目测试 计划的延伸	293	10.2.2 持续阅读	329
9.2.2 制订个人测试计划时应该综合 考虑各种项目元素	294	10.3 且行且思	330
9.2.3 测试需要动态管理	297	10.4 成为专家	333
		10.5 小结	335
		参考文献	336

第 1 章

软件测试基础

本章将阐述一些软件测试的基本观点，它们是全书的基石。虽然它们并没有得到所有人的认可，但是我认为它们反映了软件测试的基本事实。测试人员需要仔细分析它们，并用于实践，才能有效地实施软件测试。

1.1 软件的复杂度已经超越了人的理解能力

必应词典^①是我常用的一款桌面软件（如图1-1所示），它将用户输入的英文单词发送给必应服务器，然后接受服务器所返回的中文解释，最后将中译显示在界面上。



图1-1 必应词典

^① <http://dict.bing.msn.cn>.