

软件测试泰斗传道解惑  
淘宝测试技术专家翻译

Google软件测试精髓完美呈现  
测试界知名专家鼎力推荐

PEARSON



像Google一样进行软件测试

【美】James Whittaker Jason Arbon Jeff Carollo 著  
黄利 李中杰 薛明 译

测试工程师人生  
测试开发工程师人生  
谷歌员工访谈录

人民邮电出版社  
POSTS & TELECOM PRESS

014009947

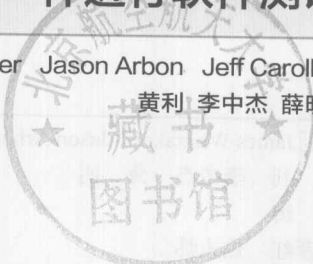
TP311.55  
02



# 软件测试之道

像Google一样进行软件测试

【美】James Whittaker Jason Arbon Jeff Carollo 著  
黄利 李中杰 薛明 译



TP 311.55  
02

人民邮电出版社  
北京

图书在版编目 (CIP) 数据

Google软件测试之道 / (美) 惠特克  
(Whittaker, J.), (美) 阿尔邦 (Arbon, J.), (美) 卡  
罗洛 (Carollo, J.) 著; 黄利, 李中杰, 薛明译. — 北  
京: 人民邮电出版社, 2013.10 (2013.11 重印)  
ISBN 978-7-115-33024-6

I. ①G… II. ①惠… ②阿… ③卡… ④黄… ⑤李…  
⑥薛… III. ①软件—测试 IV. ①TP311.5

中国版本图书馆CIP数据核字(2013)第209033号

版 权 声 明

Authorized translation from the English language edition, entitled: *How Google Tests Software*, 9780321803023 by James Whittaker, Jason Arbon, Jeff Carollo, published by Pearson Education, Inc., copyright © 2012. Pearson Education, Inc. All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

CHINESE SIMPLIFIED language edition published by PEARSON EDUCATION ASIA LTD. and POSTS & TELECOM PRESS Copyright © 2013.

本书中文简体字版由 Pearson Education Asia Ltd. 授权人民邮电出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签, 无标签者不得销售。

版权所有, 侵权必究。

- 
- ◆ 著 [美] James Whittaker Jason Arbon Jeff Carollo  
译 黄利 李中杰 薛明  
责任编辑 张涛  
责任印制 程彦红 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>  
北京鑫正大印刷有限公司印刷
  - ◆ 开本: 800×1000 1/16  
印张: 18  
字数: 335千字 2013年10月第1版  
印数: 4 001-7 000 2013年11月北京第2次印刷  
著作权合同登记号 图字: 01-2012-8860号

---

定价: 59.00元

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

反盗版热线: (010)81055315

## 对本书的赞誉

“James Whittaker 长期以来一直都能准确把握测试领域的发展脉搏，在这个云计算变革浪潮汹涌的时代，不论对 Google 员工，还是对其他任何测试人员来说，这本书都是紧跟时代、保持竞争力的必读书籍。”

—— **Sam Guckenheimer**，微软 Visual Studio 产品及战略负责人

“Google 一贯是测试领域的创新者——无论是对手工测试与自动化测试的结合、本地团队与外包资源的融合，还是近来开创性地用真实场景测试补充实验室场景测试等方面。这种对创新的渴望帮助 Google 解决了很多新问题，更好地发布了产品应用。这本书中，James Whittaker 系统地描绘了 Google 是如何在快速发展的软件测试领域取得成功的。”

—— **Doron Reuveni**，uTest CEO 及联合创始人

“这本书改变了游戏规则，从版本的每日发布到平视显示器（译注：平视显示器是一种飞行辅助仪器。飞行员透过座舱正前方组合玻璃上的光电显示装置观察舱外景物时，可以同时看到叠加在外景上的字符、图像等信息，方便随时察看飞行参数。这里指软件系统参数的集中显示面板）。James Whittaker 把计算机科学的方法应用到软件测试领域，这将成为未来软件企业的标准。本书以平实而饶有趣味的语言风格描述了 Google 在流程和技术上的创新。对每个做软件开发的人来说，这都是一本不可多得的好书。”

—— **Michael Bachman**，Google AdSense/Display 部门高级工程经理

“通过记录 Google 测试工程实践中的大量奇思妙想，作者已经把本书打造成了现代软件测试领域的圣经。”

—— **Alberto Savoia**，Google 工程总监

“如果你要在云端发布代码并尝试建立一套保证产品质量和用户满意度的策略，你必须仔细研究和思考本书中的方法。”

—— **Phil Waligora**，Salesforce.com

“James Whittaker 在测试领域是很多人的导师和灵感源泉。如果没有他的贡献，我们在测试领域不可能拥有今天这样的人才和技术。我一直敬畏他的魄力和激情。作为业界巨擘，他的作品绝对值得每位 IT 行业的人阅读。”

—— Stewart Noakes, 英国 TCL 集团总裁

“当 James Whittaker 在微软工作的时候我曾与他共事。虽然我怀念与他一起在微软的日子，但我知道他在 Google 会从事伟大的工作。这本书包含了各种创新的测试理念、实践案例及对 Google 测试体系的深刻洞察。任何对 Google 测试和质量技术稍感好奇的人，或有意发现一些崭新测试思路的人，都能从这本书中有所收获。”

—— Alan Page, 微软 Xbox, 《微软的软件测试之道》的作者

# 内 容 提 要

每天，Google 都要测试和发布数百万个源文件、亿万行的代码。数以亿计的构建动作会触发几百万次的自动化测试，并在好几十万个浏览器实例上执行。面对这些看似不可能完成的任务，谷歌是如何测试的呢？

本书从内部视角告诉你这个世界上知名的互联网公司是如何应对 21 世纪软件测试的独特挑战的。本书抓住了 Google 做测试的本质，抓住了 Google 测试这个时代最复杂软件的精华。本书描述了测试解决方案，揭示了测试架构是如何设计、实现和运行的，介绍了软件测试工程师的角色；讲解了技术测试人员应该具有的技术技能；阐述了测试工程师在产品生命周期中的职责；讲述了测试管理及在 Google 的测试历史或在主要产品上发挥了重要作用的工程师的访谈，这对那些试图建立类似 Google 的测试流程或团队的人受益很大。最后，本书还介绍了作者对于 Google 测试如何继续演进的见解、Google 乃至整个业界的测试方向的一些预言，相信很多读者都会感受到其中的洞察力，甚至感到震惊。本书可以作为任何从事软件测试人员到达目标的指南。

本书适合开发人员、测试人员、测试管理人员使用，也适合大中专院校相关专业师生的学习用书，以及培训学校的教材。



## 致中国读者

It brings me great pleasure that the demand for this book was strong enough in China to make this translation possible. China is a major player in the software industry, and I am pleased that some of my work is available to the millions of software professionals in this country. May your code have few bugs and many adoring users!

James Whittaker

“看到这本书在中国的需求如此旺盛，以及它的中译版最终付梓，真让我喜出望外，难以言表。在整个软件产业版图中，中国占据着非常重要的位置，如果说我的一些工作能给中国的软件同仁带来些许帮助，幸甚至哉。祝愿你们的代码少一些 bug，多一些挚爱的用户。”

James Whittaker

## 译者序

毫无疑问，在当前这个时代，处于浪潮之巅的伟大公司非 Google 莫属。很长一段时间以来，Google 的技术一直被外界所觊觎，其所宣扬的工程师文化氛围也成为了许多工程师梦寐以求的技术殿堂，其内部的工程实践更是技术分享大会中最热门的话题之一。但迄今为止，没有一本书系统地介绍 Google 内部产品的研发流程与模式，包括开发、测试、发布、团队成员如何分工协作等细节，直到《How Google Tests Software》的出现，才使得我们有机会管中窥豹，了解 Google 技术神秘之处。这也是我们翻译这本书的第一个原因。

正如本书中所提及的那样，互联网的出现改变了许多软件研发的模式。许多曾经红极一时的传统测试书籍里提及的最佳测试实践，在当前的环境下，效率会大大下降，在一些极端的情况下甚至会适得其反。我自己就是一名测试工程师，从事互联网方面的测试工作，对此深有体会，也经常焦虑如何在制约质量和快速发布之间寻找平衡，所以，也特别想从一些主流互联网公司的测试模式中得到启发和借鉴，特别是想看一下这个世界上最成功、增长速度最快的互联网公司——Google，是如何应对互联网测试挑战的。通过翻译这本书，自己学到了更多感兴趣的知识。这也是我们翻译这本书的第二个原因。

James Whittaker 在正式撰写本书英文版之前，于 2011 年 1 月在 Google Testing Blog 上尝试发表了“*How Google Test Software*”系列文章。当看到第一篇时我就被深深地吸引住了，第一感觉就是，太棒了！Google 测试团队居然是这样组织的！之后，随着这个系列文章的逐一公开，Google 也逐渐揭开了其神秘面纱，让我对其测试实践也有了越来越多的了解，但了解的越多，疑惑也就越多。不得不承认，这几篇文章就像正餐前的开胃小菜，它完全勾起了大家的食欲，仅仅依赖这几篇文章完全不能满足窥探 Google 测试体系的需求。在 2011 年 11 月的 GTAC (Google Test Automation Conference) 大会上，我见到了 James 本人，便聊起了《*How Google Test Software*》这本书，James 一听到又有人在打探这本书的下落，乐呵得嘴都合不拢了，却卖起了关子来，只是说书快出版了。大约在 2012 年 9 月，这本书的英文版终于问世之后，突然接到李中杰（本书的合译者之一）的电话，问我为什么不去翻译一下这本书呢。之前虽然是兴趣使然，做过那几篇文章的翻译，但与翻译一本书相比，还是有些微不足道的。但几经转辗，还是机缘巧合地去做了一件事情，这也是翻译



这本书的第三个原因吧。

最后要说的，也是最重要的一个原因。我原本根本没有这么大的勇气来完成这件事情。众所周知，James 不仅是测试领域的泰山北斗，而且他颇具文学功底，语言诙谐幽默，妙笔生花，翻译他的书籍，让我诚惶诚恐，以至于焦虑得昼夜不安。但两位合译者，李中杰博士和薛明，他们的乐观与自信让本书的翻译得以完成。与他们两位的合作，幸福之感难以言表，所收获的也不仅仅是长知识那么简单，更有许多惊喜深藏内心。翻译别人的书，像是在反刍，再精彩也是在讲别人的故事，还是期待有朝一日，能够也有机会讲讲自己的故事。

最后祝愿国内的读者能够从这本书中有所借鉴，找到适合自己现状的开发测试模式。由于译者水平有限，错漏之处在所难免，若有欠妥之处，欢迎指正。

## 《Google 软件测试之道》业界热评

“Google 的测试理念有什么与众不同？Google 的快速开发、快速发布的秘密又是什么？《Google 软件测试之道》将 Google 的测试、产品的发布变得没有那么神秘，本书系统地介绍了 Google 的测试理念、自动化测试技术、产品发布流程，以及测试团队的组成和测试工程师的招聘，是一本真心做技术分享的好书！”

—— 张南，Google 中国测试经理

“读完本书，Google 测试就像一副完美的测试画卷展现在我的面前。没错，我说的是‘完美’！测试领域一直倡导的诸多测试理念，如尽早测试、注重早期测试和评审、注重测试人员技能等，对于很多测试团队而言，是那么的理想化，以至于实施起来困难重重，而在 Google 都已化作种种测试实践，自然又现实。感谢译者的工作，让更多中国的测试人员可以从中学借鉴 Google 测试的优秀实践。”

—— 邵晓梅，独立软件测试培训与咨询顾问、首届 ChinaTest 大会执行主席

“我 2007 年刚加入 Google 中国时，就被这家企业具有的测试文化深深吸引。Google 内将测试推到上游的实践、内建质量的意识，以及优秀的自动化测试实践，无一不让我觉得兴奋。在担任 Google 中国区的测试负责人期间，我也多次向外界介绍 Google 的测试实践，希望 Google 的实践经验能够更好地帮助到更多人。James 的这本书详尽地介绍了 Google 的测试体系与测试实践，是一本即系统又非常‘接地气’的书。很高兴看到人民邮电出版社组织将这本好书翻译成中文，相信每位读者都能从本书中受益匪浅。”

—— 段念，豆瓣工程副总裁，曾任 Google 中国测试经理

“这本介绍 Google 软件工程生产力的好书值得每一位软件测试人员和研发管理者拥有，我个人甚至认为这是软件行业十年难得一遇的好书，书中所描述的观点、测试人员的价值拓展和测试技术创新实践不仅对互联网行业的软件测试从业人员有着非常好的借鉴意

义，而且也为其他行业的软件工程人员提供了‘新的翅膀’，让大家都能飞得更快、更高。正确的认知是一切成功的源头，也许你能很容易找到十个拒绝了解不同观点的理由，但你依然可以找到十个理由去接受不同的新观点，兼听则明会让你的工作更高效，自己做得更开心，过得更充实。”

——董杰，百度在线网络技术有限公司测试架构师

“测试从业者已经非常熟悉 James Whittaker 了，除了熟知的他的作品《探索式软件测试》、《How To Break Software》两本书之外，《How Google Test Software》更是大家关注已久的精品。在软件产品时代，微软公司是软件测试领域的标杆，而在今天的 SaaS 或云服务时代，Google 成了软件测试领域新的标杆，Google 在测试组织、流程和方法上大胆创新，构造了更具效率和灵活性的测试体系，特别是在持续测试、自动化测试、在线测试、开发测试等方面有着众多的优秀实践，值得我们学习、思考和借鉴。”

——朱少民，国内软件测试的领军人物，同济大学软件学院教授

“软件测试方法会产生颠覆性的变化吗？未来还需要测试工程师吗？最近一年这样的话题被持续地讨论，我没有结论，但是我觉得与其喋喋不休地争论，不如让我们看看世界级的 IT 企业 Google 是如何做测试的。通过本书让我们理解了 Google 的测试理念，理解了 Google 的工程师文化，从中你能发现更适合你的测试方法！”

——贺炘，领测国际创始人

“这本书是我推荐读者了解敏捷测试思想和技术的第一读物，没有之一。这本书的内容全部来自一线实际经验，而非理论空谈。更为重要的是，它传递了一种非常重要的理性质量观，同时还对如何将这种理性质量观落地给出了非常具体的建议。”

——吴穹，敏捷咨询师（在敏捷测试、自动化测试方面有深入研究）

“对于互联网公司，在快速前进中保持高质量是一个永恒的难题，在去哪儿网内部，开发工程师、产品经理都需要参加测试，以此来提醒——质量是所有人的事情而不只是测试团队的事情，但是，依然有太多的质量问题和实施中的难题没办法解决。本书可以给那些关注如何在此困境中突围的人们很多启发。”

——吴永强，去哪儿网 CTO

“感谢译者翻译了这本测试业内的经典之作，让国内的测试团队能够快速理解国际测试的发展并跟上国际节奏。我有幸先阅读了本书的部分内容，对 Patrick Copeland 在序中描述的测试变革的心路历程深有共鸣：招聘具备开发能力的测试人员难，找到懂测试的开发人员更难；团队的变革开发团队不接受，测试团队也不买账。同时，我们面临的挑战比 Google 更大，我们不仅要做好自动化，做好持续集成，做好测试工具，做好研发生产力，我们还要将测试技术与产品和业务结合，促进集团内产品和业务的发展。因此，与 Google 的测试人员相比，我们不仅要具备开发能力、测试思维，还要具备业务思维，能深刻理解业务所服务的客户需求及客户价值。做好工程，更要做好业务！加油！”

—— 夏林娜，阿里巴巴集团测试总监

“互联网快速响应变化的需求彻底颠覆了传统的软件开发和测试模式，敏捷、持续构建和开发自测等成为测试行业的热点话题。Google 无疑走在测试变革的最前沿，并已经在互联网领域产生广泛的影响并拥有大批拥趸。Google 的全新测试理念和组织形式非常值得国内的同行借鉴。”

—— 刘立川，阿里巴巴集团测试总监

“或许有人会质疑，互联网公司也可以有很好的测试吗？此书可能会改变他们的观点。第一，本书第一作者 James Whittaker 是一个在微软接受了最正统测试理念的人，又从互联网的视角解读测试，这让他的观点全面而具有说服力；第二，这本书的中文翻译非常出色，读起来像测试行家如数家珍。所以，我强烈推荐本书，Google 的测试不一定是出色的，但这本书是。”

—— 柴阿峰，测试圈儿里那个说相声的

“我和本书的三位作者在西雅图有很多交流，并曾经共事。James Whittaker 是软件测试界强有力的执行者、探索者和思考者。本书是他和另外两位作者在 Google 工作的全面、详细总结和提炼。他们从软件测试开发工程师、软件测试工程师以及测试经理三个不同角度出发，详细阐述了 Google 软件测试之道，给企业，特别是互联网企业在如何测试、如何保证产品质量等方面提供了很好的参考。同时开阔了我们的视野，让我们对软件测试的职责、手段和未来发展有所思考。”

—— Bill Liu, Software Design Engineer in Test, Amazon

## 关于这本书

在 Patrick Copeland 最初建议我写这本书的时候，我有些犹豫，犹豫的理由后来也被逐一证实它们确实值得思考。人们会质疑我是否是写这本书的最佳候选 Googler（他们也确实怀疑过）。有着太多的人想参与到这本书的撰写之中（后来也证实的确如此）。但更重要的是，我之前出版的一些书籍多数是给初学新手看的，像“*How to Break*”系列和《*Exploratory Testing*》，都是在从头到尾讲一个完整的故事。这本书并不是这样。读者可能坐着一口气读完，但其实它更适合作为一本参考书，一本介绍 Google 是如何完成大小规模不一的测试任务的参考书。我希望本书的读者是一些已经在公司从事测试工作的人，而不是一些初学者，他们会有一些基础，并会比较 Google 的流程与他们所使用的流程之间的区别，这样他们的收获更大。我憧憬着经验丰富的测试人员、测试经理、管理者能够随手拿起这本书，找一些感兴趣的话题，看一下在某些方面 Google 是如何做的。这可真不是我惯用的写作风格。

在此之前从没有写过书的两位工程师，为了这本书，加入进来共同努力。这两位都是优秀的工程师，他们在 Google 的工作年限都比我长。Jason Arbon 的职位是 TE（测试工程师），但他内心深处有着创业情怀，在本书“测试工程师”这一章中出现的许多工具和想法，都深受他的影响。我们有幸一起共事，并彼此从对方身上受益良多。Jeff Carollo 也是一名测试人员，但后来转做开发了。Jeff Carollo 是我见过的最优秀的那一类 SET（软件测试开发工程师），也是少数几个我认识的那种可以写出“自动化之后就不用再参与”的代码的人之一，他的测试代码写得非常棒，可以独立运行不需要任何干预。我与这两位才华横溢的人共同写作，并在风格上尽可能地达成一致。

有许多 Googler 提供了资料。当资料中的文字和标题是同一个人的工作时，我们会在标题中把这个人标记一下。还有许多对 Google 测试发挥了深刻影响的人，我们针对这些人做了一些采访。这是我们能想到的最好的、让尽可能多的曾经定义了 Google 测试的人参与进来的方法，而不是搞一本由 30 个人合著而成的书。不一定所有的读者对这些访谈都感兴趣，但在书中可以很清晰地找到这些访谈的起止位置，以便选择跳过这一部分，或者专门找到这部分来阅读。我们同样感谢为数众多的贡献者，但如果有不到之处，也愿意接受任何批评。英语实在是一门贫乏的语言，无法用它描述出这些工作是多么地卓越和辉煌。

快乐阅读，快乐测试，祝愿你总能发现（并修复）bug。

James Whittaker

Jason Arbon

Jeff Carollo

献给 Google、Microsoft 和全世界给我启发的测试人员。

——James Whittaker

献给我的妻子 Heather 和我的孩子们 Luca、Mateo、Dante 和 Odessa，他们一直认为这段时间我在星巴克工作。

——Jason Arbon

献给我的妈妈、爸爸、Lauren 和 Alex。

——Jeff Carollo



## 致 谢

我们想感谢那些不知疲倦地、致力于质量改进的 Google 工程师们。同样，也非常感谢 Google 开放的工程和管理文化，在对待测试方法与实践方面与 Google 打造其他产品如出一辙，允许不断创新以及天马行空般自由思维的存在。

在这里要特别向那些投入巨大精力并勇于承担风险将测试推向云端的人们致敬，他们是 Alexis O. Torres, Joe Muharksy, Danielle Drew, Richard Bustamante, Po Hu, Jim Reardon, Tejas Shah, Julie Ralph, Eriel Thomas, Joe Mikhail, Ibrahim El Far。还要感谢我们的编辑，Chris Guzikowski 和 Chris Zahn，他们一直非常有礼貌地在容忍我们这些工程师的唠叨。感谢那些受访者在书中分享他们的观点与经验，他们是 Ankit Mehta, Joel Hynoski, Lindsay Webster, Apple Chow, Mark Striebeck, Neal Norwitz, Tracy Bialik, Russ Rufer, Ted Mao, Shelton Mar, Ashish Kumar, Sujay Sahni, Brad Green, Simon Stewart, Hung Dang。特别要感谢一下 Alberto Savoia，他在原型及快速迭代方面的灵感成就了今日 Google 快速发布的文化。感谢 Google 餐厅的工作人员，他们提供了美味的餐饮。感谢 Phil Waligora, Alan Page, Michael Bachman，他们为本书提供了率直坦诚的反馈。最后，要特别感谢 Pat Copeland，是他将来自五湖四海且充满激情的各路精英汇集于此，并投身于质量方面的不断改进工作。

# 序

Alberto Savoia

谷歌工程总监

为一本你曾经想自己去撰写的书去做序，是一种尴尬的荣誉，这种感觉有点像你被邀请去为好友做伴郎，但新娘却是你曾经心爱的姑娘。但是 James Whittaker 却是一个聪明的家伙，在他问我是否愿意为这本书写序之前，先请我吃了一顿我非常喜欢的墨西哥晚餐，并让我喝了几杯墨西哥 Dos Equis 啤酒。当我还沉浸在牛油果酱带来的愉悦时，他终于提出了这个请求，在当时那种气氛下，我只能强作欢颜并答应了他：“没有问题。”他的诡计“得逞”了，他和他的“新娘”——这本书，一起站在一边，而我却不得不在这里为他们的婚礼做致辞。

正如我说过的，他是一个聪明的家伙。

让我继续写这篇序吧，为这本我曾想自己写的书。

这个世界上真的还需要另外一本关于软件测试的书吗？特别是 James Whittaker，这个高产的家伙，一个我曾经不止一次公开地称其为测试书籍出版界高产的“八胞胎妈妈”（译注：不知道“八胞胎妈妈（Octomom）”是什么意思？Google 一下你就知道），还需要他的这么一本软件测试书吗？那种讲述陈旧得令人厌烦的测试方法学和宣扬一些可疑、过时的建议的书还少吗？是的，这样的书已经足够多了，但我认为这本书绝非如此。这也是我想自己去写它的原因，这个世界很需要这样一本独特的测试书。

互联网的出现急剧地改变了许多软件设计、开发和发布的方式。很多曾经红极一时的测试书籍里提及的最佳测试实践，在当前的环境下效率会大大下降，或者毫无效果，甚而在某些情况下会事与愿违地起反作用。在互联网和软件产业，一切变化都如此迅速，以至于许多最近几年才出版的软件测试方面的书籍都已陈腐过时，打个比方，它们就像讲述水蛭吸血和开颅驱赶恶鬼的外科手术书一样。对付这种书，最好的办法就是直接把它们扔掉，或者做些有益的事情，例如，循环再利用，做出纸尿裤来，以防止流落到容易上当受骗的人之手。

考虑到软件产业的发展速度如此之快，如果说十年后这本书也过气了，那一点儿也不奇怪。但在下次浪潮来临之前，这本书可以既适时又适用地从内部视角告诉你这个世界上最成功、增长速度最快的互联网公司之一，是如何应对 21 世纪软件测试的独特挑战的。James Whittaker 和他的伙伴们，抓住了 Google 如何做测试的本质，抓住了 Google 如何测试我们这个时代最复杂和流行软件的精华。我之所以了解这些，是因为我从头到尾经历了这个伟大的转变。

我于 2001 年以工程总监的身份加入 Google。当时，Google 大概有 200 名开发人员，但只有区区 3 位测试人员！那个时候，开发人员已经开始做自己代码的测试了，但由于测试驱动开发的模式才刚刚开始，而且像 JUnit 这样的测试框架也没有大规模使用。当时的测试主要是在做一些随机测试 (ad-hoc testing)，其好坏取决于编写代码的开发者的责任心。但即使那样也是可以接受的，因为，当时正处在创业阶段，必须快速前进并勇于冒险，否则就无法和那个时代已经非常强大的对手竞争。

然而，当 Google 逐渐成长变大，Google 的一些产品对于最终用户和客户来说开始变得至关重要（例如，竞价广告产品，我曾经负责的产品，很快变成许多网站的主要收入来源），我们清晰地认识到必须加大对测试的关注和投入。但只有 3 个测试工程师，别无选择，只能让开发来做更多的测试。与其他的几个 Googler（译注：Google 员工，本书中一般指 Google 工程师）一起，我们介绍、培训、推行单元测试，我们鼓励开发人员把测试作为优先级较高的事去做，并建议使用一些工具，如 JUnit，把测试做成自动化的。但是进展缓慢，并非所有的人都接受、认同开发人员去做测试这件事情。为了继续保持这个势头，在每周五下午公司的啤酒狂欢时（译注：TGIF, Thank God It's Friday, Google 在每周五下午举行全员聚会），我们为一些做测试的开发人员颁发奖品来激励大家。但这种感觉不是很好，有点像杂技训兽师在小狗完成某个动作后给一些奖励一样，但这样至少还是把大家的注意力吸引到测试上了。会如此幸运吗？如此简单就可以让开发做测试了？

很不幸的是这招根本不管用。开发人员发现，为了测试充分，他们不得不针对每一行功能代码，写两到三行的单元测试代码，而且这些测试代码和功能代码一样都需要维护，且有着相同的出错概率。而且大家也意识到，仅做单元测试是不够的，仍然需要集成测试、系统测试、用户界面等方面的测试。当真正开始要去做测试的时候，会发现测试工作量变得非常大（且需要很多知识的学习），并要求在很短的时间内完成测试，要以“迅雷不及掩耳”之势完成。

我们为什么要在很短的时间内迅速地完成测试呢？我一直这么认为，对于一个坏点子或考虑欠周的产品，即便再多的测试，也无法把它变成一个成功的产品。但如果测试方法不当，却会扼杀一个本来有机会成功的产品或公司，至少会拖慢这个产品的速度，让竞争