

TURING

图灵程序设计丛书

Apress®

Pro Website Development and Operations

Streamlining DevOps for Large-Scale Websites

深入 网站开发和运维

[美] Matthew Sacks 著

曾少宁 译



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

Pro Website Development and Operations
Streamlining DevOps for Large-Scale Websites

深入 网站开发和运维

[美] Matthew Sacks 著

曾少宁 译

人民邮电出版社

北京

TP393.092

2545

图书在版编目 (C I P) 数据

深入网站开发和运维 / (美) 萨克斯 (Sacks, M.) 著;
曾少宁译. — 北京: 人民邮电出版社, 2014. 4

(图灵程序设计丛书)

书名原文: Pro website development and
operations: Streamlining devOps for large-scale
websites

ISBN 978-7-115-35023-7

I. ①深… II. ①萨… ②曾… III. ①网站—开发
IV. ①TP393.092

中国版本图书馆CIP数据核字(2014)第049920号

内 容 提 要

本书针对大型网站及移动网站开发运维的现状问题, 结合敏捷方法, 阐述了“开发运维”这一理念。作者分享了在创建和运维大型生产网站时所需要的经验, 讲述了如何综合运用系统、代码和进程来使大型网站的开发和运维变得超级高效。

本书适合网站开发人员和运维人员阅读。

◆ 著 [美] Matthew Sacks

译 曾少宁

责任编辑 丁晓昀

执行编辑 李 鑫

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>

北京天宇星印刷厂印刷

◆ 开本: 880×1230 1/32

印张: 5.5

字数: 131千字

2014年4月第1版

印数: 1—3 000册

2014年4月北京第1次印刷

著作权合同登记号 图字: 01-2012-7084号

定价: 35.00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



版 权 声 明

Original English language edition, entitled *Pro Website Development and Operations: Streamlining DevOps for Large-Scale Websites* by Matthew Sacks, published by Apress, 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705 USA.

Copyright © 2012 by Matthew Sacks. Simplified Chinese-language edition copyright © 2014 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由Apress L. P.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。

版权所有，侵权必究。

技术审稿人

Patrick Debois 多年来一直致力于消除开发与运维间的隔阂。2009 年，他组织了第一届 devopsdays.org 大会，在这次大会首次提出了“开发运维”的概念。他积极参与开发运维社会，分享了大量关于技术与文化方面的观点，并一直鼓励其他人也这么做，特别是在传统 IT 企业中推广开发运维理念，帮助人们转变意识，互相协作以取得更好的成果。

译者序

开发运维（DevOps）并不是一个新概念，如今却成为了技术社区里的一个热门词汇。开发运维包含多个概念，如“敏捷系统管理”、“敏捷运维”等。一开始，开发运维主要就是为了解决开发团队与运维团队之间的沟通问题^①。在定义上，开发运维指的是系统管理员与开发人员共同参与一个敏捷开发过程，使用相同的敏捷开发技术去完成他们的系统管理工作^②。

作者在文中提到：“开发运维代表了一种新的文化视点，它促使开发人员与运维工程师展开协作……”因此，本书介绍的开发运维基础概念和最佳实践方法适合参与网站开发过程的所有角色，包括与开发运维直接相关的开发人员和运维人员，也包括管理人员、架构师、测试人员和营销人员等。

在内容上，本书遵循具体问题具体分析的原则，许多内容都是为

① <http://dev2ops.org/2010/02/what-is-devops/>

② <http://theagileadmin.com/what-is-devops/>

了帮助读者寻找最适合自己的特殊环境的实践方法。本书具体内容如下。

- 第 1 章介绍了成功网站的开发运维原则，列举和说明了一些引领网站成功发布的最佳实践方法，以及一些专家对于开发运维的见解。
- 第 2 章介绍了改善工程人员和业务运维人员关系的实践方法，涉及文化、技术、工具和管理等多方面，另外还介绍了管理层在改善人员沟通方面的正确做法。
- 第 3 章详细介绍了 Web 测试实践方法和步骤，指导读者基于行为驱动的开发方法实现自动化 Web 测试，并且说明了如何确定测试对象及测试目标。
- 第 4 章说明了开发运维中常见的文档问题及解决方法，同时举例说明了各种文档类型及模板，如 API 文档、入门指南、用例文档、交互流程文档、架构图和基础架构设计文档等。
- 第 5 章介绍了实现基础架构和应用程序自动分配的实践方法，在回顾 Web 应用的各个层次构成之后，通过一个自动化案例说明了该如何选择基础架构管理与分配框架。
- 第 6 章从整体上介绍了发布网站所经历的各个阶段，以及各个阶段常出现的运维问题及解决方法。
- 第 7 章扩展介绍了如何在整合移动 Web 之后实现与原生应用保持一致的用户体验。

作者在开发运维方面有丰富的工作经历，积累了许多宝贵的实践经验，对于开发运维的一些关键问题有着独到的见解。例如，在介绍如何改善 IT 人员与业务人员沟通的内容中，作者写到当今企业或单位内部常见的内部矛盾形式，语言很犀利！

个人感觉作者的母语并不是英语（有待求证），其用词习惯与母语为英语的作者有所不同。在翻译过程中，我尽了自己最大努力做到忠实原文，尽量把作者的见解表达到位。但是，由于本人能力所限，译文中难免有所错漏，请多多包涵！

欢迎指正！

曾少宁

2013年12月19日 写于广东惠州

序

一些问题我们都曾遇到过：长研发周期、复杂流程、部署失败、推倒、重做、半夜报警、“在我的主机上好好的，怎么到这儿就不行了”、紧急更新，等等，所有问题都夹杂着责骂声袭来。所有参与过生产软件应用程序的开发或维护的人，几乎都有过相似的经历。

我们相信一定有一种更好的方法。

开发运维所蕴含的概念其实非常简单：将有不同技能的人团结起来，扫除成功路上的所有拦路虎，一起完成一个共同的目标。敏捷方法广泛而成功地改善了业务人员与技术人员间的沟通，而开发运维理念的基础正是敏捷方法。开发运维将相同的理念应用到了传统上相互独立的开发、QA 和运维团队上。

开发运维方法始于 2009 年，在 2010 年发展成熟。随着人们对它的了解和实践的增多，它背后的理念一定会产生重要的影响。

此外，最近几年迅猛发展的技术因素促使开发人员和系统管理员更

紧密地协作，这也加速了开发运维的发展。正如 Patrick Debois 所说的：“虚拟化使运维人员能够快速创建新环境，云计算则完全解决了资源问题。开发运维的真正特点源于两个概念：配置管理和基础架构即代码。”^①

对于这种方法的发展壮大，我并不感到意外。多年来，我的工作就是改进应用的交付过程，实现项目技术知识与反馈的集中管理，所以，当第一次听说“开发运维”这一概念及其要解决的问题时，我马上就体会到了它的含义。这就像是一个“触电”时刻。

对于业务而言，减少或消除开发人员和运维人员之间的隔阂是非常重要的，因为他们不仅需要尽快地将新应用和特性交付市场，也需要在运维中保证它们有更符合现代互联网要求的稳定性和规模。

虽然概念简单，但是它给大多数组织带来了一种巨大的文化挑战，特别是那些大型或流程已根深蒂固的组织。而且，即使在通常情况下开发运维解决的问题还算比较清晰，我们仍然需要做很多工作，才能理解它对于组织的真正意义。和敏捷的情况一样，针对开发运维实践的“正确方式”仍然有些争议，不同的团队在努力实现自己的计划时仍然会产生分歧。但是，它的基本理念是无可争议的。

本书关注一些基础概念，读者可以用它们来确定团队内部、团队之间存在的问题，还会给出一些解决问题的最佳实践方法。本书遵循具体问题具体分析的原则，因此不会介绍某一种特定的方法，而是帮助读者寻找适用于自身特殊环境的观点和方法，并且了解在这个过程中可能遇到的挑战和得到的结果。

^① Cutter IT Journal，第 24 卷第 8 期，2011 年 8 月。

此外，本书还会介绍一些来自知名企业的专家经历，了解他们曾面对过与战胜过的困难。

亲爱的读者，我真心希望你能够在自己的组织中运用本书所介绍的理念。

Brett Porter, MaestroDev CTO

致 谢

感谢我的家人和女朋友。没有你们的耐心和支持，容忍我连续熬夜工作，我不可能完成本书的创作。同时也要特别感谢我的导师 Safdar Husain、Blake Swopes 和 John Martin。你们的悉心教导使我在工作中慢慢成长，让我的处事效率越来越高。此外，感谢 USENIX 和 Apache 软件基金会，允许我为你们组织贡献尽绵薄之力。最后，我还要感谢购买本书的读者。

特别感谢参与本书编著工作的人：Tom Limoncelli、Rik Farrow、Paddy Hannon、Aslakey Hellesoy、Santiago Suarez Ordoñez、Brett Porter 及所有助我出版本书的人。

目录 CONTENTS

第1章 成功网站的开发运维原则 / 1

1.1 深入理解网站开发与运维 / 3

1.1.1 展开协作 / 7

1.1.2 提升效率 / 10

1.1.3 推进协作 / 11

1.1.4 处理变更 / 13

1.1.5 展望未来 / 17

1.2 来自专家的见解 / 18

1.2.1 软件工程师眼里的开发运维 / 18

1.2.2 运维工程师眼里的开发运维 / 21

1.3 本章小结 / 22

第2章 协调开发人员与业务运维人员 / 25

2.1 平衡技术人员与业务人员的关系 / 26

2 | 深入网站开发和运维

- 2.2 理解开发者文化 /27
 - 2.2.1 专业技能分类 /29
 - 2.2.2 人才与激励 /30
- 2.3 业务人员与 IT 人员之间的健康关系 /31
 - 2.3.1 业务人员要理解技术能力 /31
 - 2.3.2 业务成功符合工程人员的既得利益 /32
 - 2.3.3 业务人员与 IT 人员要相互理解 /32
 - 2.3.4 业务管理层要让 IT 人员参与决策制定过程 /32
 - 2.3.5 利用工具形成通用词汇表 /34
 - 2.3.6 跑在截止日期之前 /35
 - 2.3.7 释放紧张环境的压力 /36
 - 2.3.8 让业务部门感受到激励 /36
- 2.4 内部敌人 /36
 - 2.4.1 认清形势 /37
 - 2.4.2 向主管提建议可能很困难 /38
 - 2.4.3 突破界限 /39
- 2.5 改善业务人员与工程人员的沟通 /40
 - 2.5.1 定义与执行 /41
 - 2.5.2 广开言路 /41
- 2.6 本章小结 /42

第3章 Web 测试实践 /43

- 3.1 Web 测试实践 /45
 - 3.1.1 最大容量测试 /46
 - 3.1.2 持续负载测试 /47
- 3.2 行为驱动开发 /48
 - 3.2.1 Santiago Suarez Ordoñez 的自动化 Web 测试 /52
 - 3.2.2 将安全性视为测试实践 /54
- 3.3 确定测试对象 /54
- 3.4 指标吻合度测试：确定测试目标 /55
 - 3.4.1 网站的业务指标 /56

- 3.4.2 Web 应用性能指标 /60
- 3.4.3 基于指标模板的应用程序性能指标监控实践 /60
- 3.5 测试各个组件，快速修复故障 /63
- 3.6 逐层保存历史性能数据 /65
- 3.7 本章小结 /68

第 4 章 设计聪明的文档 / 69

- 4.1 鲜为人知的文档优点 /70
- 4.2 文档编写的拦路虎 /72
 - 4.2.1 场景 1：时间不够 /72
 - 4.2.2 场景 2：只有技术文档 /74
 - 4.2.3 场景 3：文档很快会过时 /76
- 4.3 文档类型与模板 /77
 - 4.3.1 API 规范与参考文档 /78
 - 4.3.2 入门指南 /80
 - 4.3.3 用例文档 /82
 - 4.3.4 用户交互工作流程 /84
 - 4.3.5 架构图 /85
 - 4.3.6 基础架构设计文档 /87
- 4.4 自动化文档编写 /89
- 4.5 本章小结 /89

第 5 章 基础架构与应用程序的自动分配 / 91

- 5.1 回顾 Web 层次 /92
 - 5.1.1 通过自动化实现统一的 Web 环境 /93
 - 5.1.2 提前计算自动化的效果 /95
- 5.2 选择自动化流程 /97
- 5.3 一个自动化场景 /101
- 5.4 降低复杂性 /104
- 5.5 选择配置管理和分配框架 /105

- 5.6 审核基础架构 /106
- 5.7 使用配置管理系统实现自动部署 /108
- 5.8 本章小结 /109

第6章 生产发布 /111

- 6.1 认识整个过程 /112
- 6.2 网站的概念开发：概念设计 /113
 - 6.2.1 概念设计要解决的基本问题 /114
 - 6.2.2 成本/收益分析 /116
 - 6.2.3 特别项目团队 /117
- 6.3 营销 /118
- 6.4 设计发布元素 /119
- 6.5 灵感与愿景 /122
- 6.6 开发 /123
 - 6.6.1 处理偏离计划的问题 /123
 - 6.6.2 研发性网站 /124
- 6.7 测试 /124
 - 6.7.1 最终用户测试 /125
 - 6.7.2 性能测试 /125
 - 6.7.3 代码分析 /128
 - 6.7.4 生产环境测试 /129
 - 6.7.5 安全测试 /131
 - 6.7.6 携带负载的压力测试 /132
- 6.8 发布前阶段 /134
- 6.9 发布阶段的负面因素：常见的人员问题 /135
 - 6.9.1 不够重视 /136
 - 6.9.2 人手不足 /136
 - 6.9.3 睡眠不足 /137
- 6.10 成功发布：防止耗竭 /138
 - 6.10.1 专责团队 /138
 - 6.10.2 特别项目团队的轮转 /139

- 6.10.3 准备应付最坏情况 /140
- 6.10.4 在产品发布过程中保持旺盛斗志 /140
- 6.10.5 成功标准 /141
- 6.11 实现一次成功的产品发布 /142
- 6.12 本章小结 /143

第7章 整合移动 Web / 145

- 7.1 为不同设备提供不同体验 /146
- 7.2 移动 Web 的局限性与用户期望 /147
 - 7.2.1 用户无耐心 /147
 - 7.2.2 延迟时间长 /148
- 7.3 理解使用模式 /149
- 7.4 对比原生应用和移动 Web 应用 /152
- 7.5 创建一致体验 /153
 - 7.5.1 转换服务 /154
 - 7.5.2 整合原生应用 /155
 - 7.5.3 通过 API 整合 /155
- 7.6 跟踪 API 使用率 /157
- 7.7 本章小结 /158