

TURING

图灵程序设计丛书

Ajax系列



MANNING

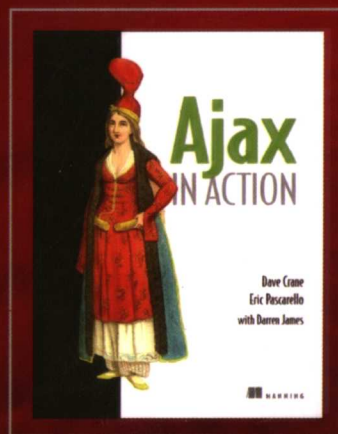
Ajax in Action

Ajax 实战

雄踞
Amazon
计算机图书榜首
数月!

Dave Crane
Eric Pascarello 著
Darren James
ajaxcn.org 译

- Ajax 技术圣经级著作!
- 全面, 深刻, 细致
- 平台无关, 适合 .NET/Java/
PHP 程序员



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

Ajax系列



Ajax 实战

Ajax in Action

Dave Crane
Eric Pascarello 著
Darren James
ajaxcn.org 译

 MANNING

 人民邮电出版社
POSTS & TELECOM PRESS

图书在版编目 (CIP) 数据

Ajax 实战 / (英) 克拉恩, 帕斯卡雷洛, 杰姆斯著; ajaxcn.org 译.

—北京: 人民邮电出版社, 2006.4

(图灵程序设计丛书)

ISBN 7-115-14717-5

I. A... II. ①克...②帕...③杰...④a... III. 计算机网络—程序设计 IV. TP393.09

中国版本图书馆 CIP 数据核字 (2006) 第 033809 号

内 容 提 要

本书是目前 Ajax 领域最为全面深入的一本著作, 其中不仅有对于基础知识的介绍, 还有对于 Ajax 开发中重大架构问题的深入探讨, 总结了大量 Ajax 开发中的设计模式, 并讨论了框架、安全性与性能等。书中提供了几个典型的例子, 兼顾各种开发平台, 这些例子的代码稍作修改就可以直接应用于项目开发之中, 代码源文件可以从图灵网站下载。

本书适用于各个层次的 Web 应用开发人员。

图灵程序设计丛书

Ajax 实战

◆ 著 Dave Crane Eric Pascarello Darren James
译 ajaxcn.org
责任编辑 傅志红

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销

◆ 开本: 800×1000 1/16
印张: 32.75
字数: 748 千字 2006 年 4 月第 1 版
印数: 1—8 000 册 2006 年 4 月北京第 1 次印刷
著作权合同登记号 图字: 01-2006-0317 号

ISBN 7-115-14717-5/TP · 5374

定价: 69.00 元

读者服务热线: (010)88593802 印装质量热线: (010)67129223

译者简介

李锐 (网名 dlee)

独立软件咨询顾问, 有多年软件开发和软件项目管理的经验, 在 J2EE、Ajax、敏捷软件开发方法等领域有独到的见解。从事软件行业 10 年来, 曾经担任过电信软件工程师、网站开发人员、售前工程师、软件项目经理、公司负责人等不同的角色, 有超过三年的 Ajax 开发经验。在 2005 年 11 月创建了 Ajax 中国网站 (ajaxcn.org), 担任站长。

赵东炜 (网名 jackyz)

Web 开发领域的一名老兵, 目前对 Java、Ajax 和 XUL 感兴趣, 除了背包旅行之外, 他也喜欢羽毛球。

王锡平

架构师, 现就职于上海一家软件公司, 具有多年企业应用软件的开发和管理经验, 但仍不满足于目前的技术发展, 主要专注于 Java、Linux 及其他新技术在企业应用软件上的使用。

我将这本书献给我的爱人和我的父母, 你们的支持就是我前进的动力!

张祖良 (网名 aliang)

专注于 Web 应用领域技术, 有 Windows DNA/.NET 和 J2EE 技术架构下开发的经历, 曾经担任过 Web 开发工程师、系统工程师和电子商务运营主管等职位。热衷于开源技术研究, 参与过 PHP Prado 项目翻译。目前比较关心 Ajax、SOA 技术和基于 Notes 企业内部管理流程的应用开发。

郑帅

中国科学技术大学学士 (计算机专业), 南洋理工大学硕士 (通讯软件与网络专业), 曾任汇丰集团 (新加坡) 高级系统分析员, 现任职于新加坡 JustCommodity 技术有限公司, 产品开发经理、系统架构师 (SCEA, OCP, ICSD), 一贯致力于将最先进的开源产品与开发思想融入实际产品的设计中, 有丰富的 J2EE 轻量级开发与设计经验, 对自己一直编程到 50 岁充满信心和期待。

张凯峰 (网名大阿福)

2004 年 4 月毕业于华北电力大学 (北京) 计算机系, 现为北大方正电子有限公司软件研发工程师, 从事媒体行业应用集成系统的开发, 热衷于技术交流与共享。

许劲松

硕士, 关注的领域有企业软件开发过程、面向服务的企业软件架构、基于 Ajax 技术的软件构件开发框架, 主要从事企业综合信息系统的分析、设计和开发。

版 权 声 明

Original English language edition published by Manning Publications Co., 209 Bruce Park Avenue, Greenwich, Connecticut 06830. Copyright © 2005 by Manning Publications Co..

Simplified Chinese-language edition copyright © 2005 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由Manning Publications Co.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。

大匠的杰作

(译者序)

普遍来说，中国人缺乏匠人的气质。“匠人”这个词在中国几乎是一个贬义词，例如说：“某某不过只是一个匠人。”但是其实国内的软件行业正是缺乏一大批对工作精益求精的软件工匠，这才是国内软件行业的主要问题所在。

相比之下，日本人是一个非常有匠人气质的民族。手艺精湛的匠人自古以来在日本都受到普遍的尊重。在这个方面，我们需要向日本人学习的地方有很多。

自从 Pete McBreen 所著的 *Software Craftsmanship* (中文版名为《软件工艺》，人民邮电出版社出版) 出版后，“软件工艺”为软件行业的从业者带来了“软件工程”之外的另一个重要的隐喻。在此之前，“软件工程”这个隐喻在软件行业中一直占据着绝对的话语权。软件工艺并非是为了彻底颠覆软件工程，而是为了促使大家将目光更多地投向精研软件建造过程中的工艺问题。这本书将软件开发者分成了三个层次：学徒开发者 (Apprentice Developer)、技师开发者 (Journeyman Developer) 和软件工匠 (Craftsman)。

Ajax in Action 的作者无疑已经达到了软件工匠的层次，本书充分体现出了他们的技术水平。我将本书评价为大匠的杰作。时间将可以证明，本书在 Ajax 技术著作的殿堂中，会像米开朗基罗的大卫那样成为不朽之作。本书就像大卫那样充满了美感，其架构的复杂、计算的精确令人敬畏，堪称是工艺学和艺术的完美结合。

想要开山立派，必须高屋建瓴。达摩祖师、张三丰无不是这样的人物。本书作为 Ajax 技术领域的开山之作，同样也达到了这样的高度。虽然它是 Ajax 领域的第一本著作，但是它所达到的高度至今仍然是同类著作难以企及的。

某个技术领域的第一本著作往往并不是该领域中最好的著作，例如，大量的开发实践已经证明早年那些介绍 J2EE 开发的书籍价值是有限的，直到 Rod Johnson 写出了经典的 *Expert J2EE Development without EJB*，对于 EJB 的盲目推崇才宣告结束。到目前为止，与 Ajax 技术相关的著作在国外已经出版了 10 本以上，国内也在今年开始陆续出版大量相关的著作，包括翻译和原创。然而，我们会发现，本书目前仍然是这个领域最为全面和深入的一本，不仅包含细致介绍的基础知识和精心设计的实战例子，更重要的是，还涵盖了对 Ajax 开发中重大架构问题的深入探讨，并且总结出了大量的 Ajax 开发设计模式，因此，认为本书是一本讲架构和设计模式的书毫不过分。

Ajax 领域的新框架和组件库层出不穷,一些功能非常简单,一些则是过度设计的产物或者存在着严重的设计问题。Ajax 开发者对于应该选择什么样的框架感到茫然无助。毫无疑问, *Ajax in Action* 可以帮助你。本书是目前已经出版的唯一一本深入探讨 Ajax 开发中的架构问题的著作,这使得它显得卓而不群。如果不去深入研究 Ajax 开发中的架构问题, Ajax 开发领域将会再现 J2EE 开发领域早年的那种混乱局面,本书将会使得 Ajax 开发秩序井然。我们需要的不完全是技术本身,还需要思考和选择的眼光,本书正是能够带给读者选择的眼光的著作。它使读者能够全方位地理解 Ajax 开发,不仅仅是看到众多的树木,更重要地是看到整片的森林。架构设计的合理性和灵活性将是选择 Ajax 框架的一个首要因素,将能决定任何 Ajax 框架未来的命运。一年之后,存在架构设计问题的很多框架都将会退出历史舞台。

尽管如此,作者并不是只讨论重大的架构问题,而忽略了具体的实现细节,这样往往会流于空泛。作者不仅仅是卓越的评论家,更是卓越的实干家。对于书中介绍的每一个架构和模式都给出了精心设计的例子。对于 Ajax 开发需要的所有基础知识,作者都通过例子做了耐心细致的讲解。书中花费了一半的篇幅阐述了大量 Ajax 开发的实例,展示了 Ajax 开发的方方面面。每一个实例中的代码最后都被重构为一个可以重用的组件,这些组件的不需要做任何修改,或者只需要稍作修改就可以直接应用在读者的项目开发之中,帮助用户得到更好的交互体验。

在本书中,有令人高山仰止的架构和设计模式的分析,有潺潺流水式的深入细致的实例讲解。娓娓道来,带给读者 100% 一对一的阅读体验。虽然讨论的内容很多,但是达到了形散神不散的效果。这个神就是改善 Web 应用的可用性,可以说,本书中所有关注的问题都是与改善 Web 应用的可用性相关的。这正是 Ajax 技术诞生的使命,也应该是任何一个 Ajax 开发者需要牢记的。Ajax 开发并不是一个纯粹的技术问题,因为这里还涉及另一个重要的利益相关人,就是最终用户。我们始终要把最终用户的感受放在首要位置,才能开发出出色的 Ajax 应用。

本书因为其内容的广泛和深入,同时适合于 Ajax 开发的初学者和提高者,两种类型的开发者都能从中得到巨大的收获。对于 Ajax 开发者来说,本书可以作为案头常备的著作。作者就像是一位吹着笛子的魔法师,循序渐进、深入浅出地将读者领进了 Ajax 这个迷人的殿堂。本书堪称是一片富饶的宝藏,等待着你——勇敢的探险家去探索 and 发现。

本书由 Ajax 中国网站 (ajaxcn.org) 组织翻译。前言部分、第 7 章和第 12 章由王锡平翻译,第 1 章~第 3 章由赵东炜翻译,第 4 章和附录 B、C 由李锐翻译,第 5 章和第 11 章由张凯峰翻译,第 6 章由许劲松翻译,第 8 章~第 10 章由张祖良翻译,第 13 章和附录 A 由郑帅翻译,全书由李锐统稿、润色及审校。

由于时间仓促,而且译者水平有限,译文中难免会出现一些错误,欢迎读者批评指正。

李锐

2006 年 4 月 4 日于上海

preface 前言

有时候，在你发觉之前，命运之神已经降临多时了。20世纪90年代初期，在我使用的各种有趣的新技术之中，一种称为JavaScript的小脚本语言刚刚崭露头角。我很快意识到，尽管名字叫JavaScript，它其实与我心爱的Java没有任何关系，但是从此它却与我结下了不解之缘。

上世纪90年代后期，我开始为数字机顶盒技术的早期采用者开发软件。其中主要软件部分的用户界面完全使用JavaScript编写，我成为一个开发小组的技术负责人，负责使用这种语言编写窗口管理代码、调度程序和各种巧妙代码。“多么神奇啊！”我想，“它总是让人意想不到。”

此后，我转向了要求更高的工作——为会说话的未来智能家居开发企业消息中枢（backbone）和各种用户界面组件。我是靠Java方面的技能被录用的，但是很快我又在编写奇妙的JavaScript用户界面了。令人惊奇的是，我发现一些人居然郑重其事地在用这种脚本语言来编写框架。很快，我使用了Mike Foster开发的早期版本的x库（你将在本书偶尔看到它的身影）。一天下午，在开发一个邮件和文本信息论坛时，我有了一个不可思议、令人兴奋的想法：在隐藏帧（hidden frame）中检查新的消息，无需刷新屏幕就可以将消息添加到用户界面中。经过几个小时的狂热编码，我成功了，甚至解决了如何用颜色来呈现新的消息的问题，使得它们容易被用户注意到。“多有趣啊！”我想，随后就又回去编写那些正经代码了。我还不知道，在同一时间，Eric Costello、Erik Hatcher、Brent Ashley和其他人正在考虑相同的事情，而微软也正在为其Outlook Web Access设计XMLHttpRequest。

命运之神还在紧随着我的脚步。我的下一个工作可以说是责任重大——为顶级的大型银行开发软件。我们混合使用了Java和JavaScript，并且使用了隐藏帧和其他的技巧。我的团队现在维护着超过1.5兆字节的代码——除了使用由JSP生成的代码外，全部是静态的JavaScript，其中还没有算上图片资源。我们用它来开发上百个操作员使用的一个应用，这个应用管理着价值数百万美元的账户。你的银行账户有可能就是使用这个软件来管理的。

在此期间，我没有完全意识到，JavaScript已经长大成人了。在2005年2月，Jesse James Garrett提供了完整拼图中缺失的一块——他为我们过去几年偷偷摸摸地私自使用的“跨浏览器异步富客户端动态HTML客户服务器技术”（cross-browser-asynchronous-rich-client-dynamic-HTML-client-server technology）取了一个简洁而干脆的名字：Ajax。

正如他们所说的，以后的一切已经载入史册。Ajax现在正在引起广泛关注。Prototype、Rico、

Dojo、qooxdoo、Sarissa和许多其他框架的开发者正在编写大量优秀的代码。框架已经多得不可胜数，但是我们还是决定尝试一下。在附录C中，我们列举了大多数值得关注的框架。生活如此美好，我从来没有像现在这样享受过计算机工作。

我们远未大功告成，这个领域还在发展之中。当我在 2005 年 9 月对 5 月撰写的第 1 章进行最后修改时，我吃惊地看到这个领域取得的巨大进展。这个主题仍然有很多值得思考的地方，接下来两年将会令人无比兴奋。在写作本书的旅程中，我非常幸运地有 Eric 和 Darren 一路相伴。

我希望你也会与我们同行——而且感到不虚此行！

Dava Crane

acknowledgments 致谢

虽然本书的封面上只有三个名字，但是有一大批有才能的、不辞辛劳的和狂热的人们在背后支持着我们。我们要感谢 Manning 出版社的每个人，特别是出版人 Marjan Bace、开发编辑 Jackie Carter 和 Doug Bennett，感谢他们对原稿众多方面的持续支持和帮助。感谢 Manning 出版社在这个项目的不同阶段与我们共事过的其他人——Blaise Bace、审稿编辑 Karen Tegtmeier、Web 管理员 Iain Shigeoka、营销专员 Helen Trimes，以及相当重要的项目编辑 Mary Piergies——感谢她引导我们如此顺利地前进。感谢文字编辑 Linda Recktenwald 和 Liz Welch、校对 Barbara Mirecki 和 Tiffany Taylor，她们的工作是不可缺少的。感谢设计编辑 Dottie Marsico 和排版 Denis Dalinnik 所做的非凡工作，将我们的涂鸦变成了图画，将一堆文字变成了一本真正的书！

很多有才能的程序员作为技术审校，毫无吝惜地为我们的工作贡献出了他们的时间。负责这项任务的是 Phil McCarthy（他不仅为我们纠正了代码，而且纠正了语法和风格，甚至使用军舰的规章使我们没有偏离航向）和 Bear Bibeault（他勇敢地在飓风“丽塔”肆虐期间还提出了许多服务器架构、客户端代码和 Mac 兼容性方面的建议）。感谢 Joe Mayo、Valentin Crettaz、James Tikalsky、Shane Witbeck、Frank Zammetti、Joel Webber、Jonathan Esterhazy、Garret Wilson、Joe Walker 和 J. B. Rainsberger 所提供的非常及时的第一流技术支持。我们真心感谢他们。

我们也感谢本书各版书稿的审阅者，感谢他们见解深刻的反馈意见：Ernest Friedman-Hill、Craig Walls、Patrick Peak、J. B. Rainsberger、Jack Herrington、Erik Hatcher、Oliver Zeigermann、Suresh Kumar、Mark Chaimungkalanont、Doug Warren、Deiveehan Nallazhagappan、Norman Richards、Mark Eagle、Christophe Avare、Bill Lynch、Wayland Chan、Shane Witbeck、Mike Stenhouse、Frank Zammetti、Brendan Murray、Ryan Cox、Valentin Crettaz、Thomas Baekdal、Peter-Paul Koch、Venkatt Guhesan、Frank Jania、Mike Foster、Bear Bibeault、Peter George、Joel Webber、Nikhil Narayana、Harshad Oak 和 Bas Vodde。

感谢 Paul Hobbs、Bill Gathen 和 Charlie Arehart 指出在本书初稿的代码中许多拼写错误。最后，特别感谢 Brian J. Sletten、Ben Galbraith 和 Kito Mann 帮助，本书的出版工作得以启动。我们也感谢本书中用到的很多 Ajax 框架的作者，并且感谢 Jesse James Garrett 为我们提供了一个简洁、干脆的缩写词，使得本书的封面显得如此优雅。（“各种富客户端 JavaScript 网络技术实战”这样的书名显然不会给人留下深刻印象）。

我们有幸站在整个巨人团队的肩上，风景是如此的美好。

DAVE CRANE

我要感谢Chia、Ben和Sophie的支持、智慧和热心，感谢她们在这段时间忍受我在所有的深夜和清晨不停工作。现在我的工作完成了，我承诺将恢复正常的生活。我也要感谢我的父母首先听说了有关写书的消息，感谢他们逐渐灌输给我力量和信心来承担这个项目。

Eric和Darren是与我共事的卓越的合著者，我也想要向他们表达我的感谢，感谢他们对本书做出的无价贡献。

感谢我在Smartstream Technologies的同事，在Ajax被命名之前与我一起探索Ajax的世界。他们是Tony Coombes、John Kellett、Phil McCarthy、Anthony Warner、Jon Green、Rob Golder、David Higgins、Owen Rees-Hayward、Greg Nwosu、Hristo Gramatikov和Stuart Martin。感谢我的经理Colin Reid和Andrew Elmore；感谢我们海外的同事Bhupendra、Pooja、Rahul、Dhiraj、Josef、Vjeko和Ted；感谢与我愉快共事多年的很多其他有才能的人们。特别感谢Rodrigo Barnes，他在十年前将一种叫做“Java”的新型编程语言介绍给我，以及我的哥哥Mike，他帮助我解决了如何使用BBC微型计算机。

ERIC PASCARELLO

感谢我的妻子Shona忍受了我在这段时间熬夜工作，感谢她在我写书期间，在没有我的帮助下筹办了我们的婚礼。感谢我的父母让我成为电脑迷。感谢我的同事Fred Grau、Paul Fuseyamore、Tim Stanton、Tracey Baker、Adrienne Cantler和Kelly Singleton忍受了我在长夜写书之后清晨的暴躁脾气；感谢www.JavaRanch.com同仁的支持和很多好主意。我也不能忘记感谢吸引我和教我编程的那些“外星人”。

DARREN JAMES

感谢我的妻子 Alana 和我的孩子 Hannah 和 Paul，他们成为了我生活中的灵感之源。感谢我的父母鼓励我，使我在学校中取得优异的成绩。感谢我的同事和朋友 Bill Scott 的好主意和支持；感谢 Richard Cowin 以及 Rico 的贡献者们；感谢 Butch Clarke，他是我在风雨中的支柱；感谢 Gordon、Junior 和 Jub-Jub 经常带来笑声和欢乐。

about this book 关于本书

在本书的写作期间，Ajax 还是一种正在成长的新技术。我们很高兴能以 Manning 出版社的实战 (*In Action*) 系列书籍的独特风格来展现它的内幕。然而在写作过程中，我们遇到了一个有趣的问题：虽然 Ajax 的热门毋庸置疑，但是它并不是真的很新，甚至不是一种真正的技术。

让我们来解释一下。Ajax 结合了几种沿用已久的 Web 技术，以新的、有趣的方式来使用它们。从某些方面来看，第一次学习使用一种全新的技术更加简单，因为你是从一张白纸开始的。Ajax 就不一样了：除了学习新的东西，还需要忘掉很多旧的东西。因为这个原因，本书和 Manning 出版社的其他 *In Action* 系列书籍的内容组织方式有些不同。在阅读的过程中，你可能会注意到这一点，并且应该知道这是我们感觉最适合这个主题的组织方式。

正如你将会看到的那样，虽然 Ajax 技术本身全部是在客户端，但是这种（以不同方式使用这种技术的）差别扩展到了服务器端。本书主要讲述客户端编程，这里大部分的代码例子都是用 JavaScript 写的。Ajax 的原则漂亮地实现了客户端与服务器端的解耦，并且能够与任何服务器端语言共同使用。因此我们拥有了广大的读者群，并且我们选择混合使用多种语言来展现服务器端的代码：PHP、Java、C# 和 Visual Basic.NET。更重要的是，尽管如此，我们还是尽量使服务器端的代码相对简单且与特定实现无关，以便你能够将代码移植到所选择的任何环境中。在确实使用了特定于语言特征的地方，我们会为不熟悉那种特殊环境的读者们进行足够详细的解释，使得他们明白我们正在做什么。

本书读者对象

Ajax 处在很多学科的交叉点上，读者可以从很多方向殊途同归。一方面，它适用于拥有计算机科学学位和数年大型软件项目实践经验的专业企业应用开发者，有时候他们需要从城堡的垛口上探出头来，做一些表示层的工作。另一个方面，它适用于已经从图形设计转向 Web 设计和“新媒体”¹的专业创意设计人员，他们自学了如何使用诸如 PHP、Visual Basic 或者 JavaScript/ActionScript 之类的脚本语言来编程。处在两者之间的读者可以是正在接受 Web 开发再培训的桌面应用开发者、想要集成多种基于 Web 管理工具的系统管理员，以及其他很多人。

1. Web 被视做一种新型的大众传播媒体。——译者注

所有这些可能成为读者的人都对 Ajax 抱有真正的兴趣。至少从某种程度上来说，我们在本书中尽力做到了满足所有人的需求。我们为习惯于把 Web 浏览器视作哑终端的服务器端开发者提供了基础 Web 技术的知识，也为可能更习惯于随手编码风格¹的新媒体的开发者提供了有关软件设计和组织的基础知识。无论你是来自哪个方面，Ajax 都是一种跨学科的技术，都将会带领你进入一些不熟悉的领域。我们将会给你一些压力，要求你在一路上学习各种新技术。我们在自己使用 Ajax 的过程中，甚至是在写作本书的过程中，也同样如此。我们发现这是一种非常有益的和令人愉快的经验，它的好处可以扩展到我们职业的其他方面。

本书的结构

本书分为四个部分。第一部分告诉你 Ajax 是什么，解释为什么它是你开发工具箱的有用补充，并且介绍能够使你取得成功的工具。第二部分涵盖了 Ajax 的核心技术，第三部分基于这些技术，讨论了从概念验证（proof of concept）到产品级的软件需要些什么条件。在第四部分中，我们采用直接的实践方法，逐步地建造了 5 个 Ajax 项目，然后将它们重构成可以在你自己的 Web 应用中使用的组件。

正如我们前面所说的，Ajax 其实不是一种技术而是一个过程。因此，第 1 章是专门为熟悉 Ajax 之前的 Web 开发方式并且正在接受再教育以适应 Ajax 的开发者而写的。我们讨论了 Ajax 应用和传统 Web 应用之间基本的不同点，如何考虑可用性，以及其他吸引人的概念。如果你想要了解有关 Ajax 的词汇是什么意思，我们建议你从这里开始。如果你仅仅想要寻找一些可以立即使用的 Ajax 代码，那么最好直接转到第 2 章。

Ajax 技术自身已经拥有了相当好的文档。在第 2 章中，我们使用例子驱动的方式蜻蜓点水般地介绍这些技术。但是我们的目标并不是无所不包，我们是要强调作为 Ajax 的一部分，哪些地方这些技术的用法与传统的开发方式不同，或者在哪些地方它们的行为与传统的开发方式是不同的。

第 3 章介绍了本书的第三个主题：管理 Ajax 代码库。亲眼见证了 JavaScript 代码库发展到超过 1.5MB，我们能够证明一个事实：为 Ajax 编写 JavaScript 与传统的开发方式相比截然不同。在这里讨论了设计模式和重构，不是因为我们认为它们很酷，而是因为我们发现使用 Ajax 时，它们是无价的、实用的工具。而且我们认为一旦上手，你也将会同意见这一点。

在第 4 章和第 5 章中，我们将视线转向 Ajax 的核心部分，并且应用设计模式的知识来发现 Ajax 的最佳实践。第 4 章考察了如何以新的方式应用旧的 Web 开发主力技术：模型-视图-控制器（MVC）模式，来使代码在客户端自身保持清晰。第 5 章考察了客户端和服务器之间通信的不同方式，以及各种不同类型的框架如何为使用 Ajax 而调整。至此，我们已经涵盖了所有的基础知识，你将会完全明白 Ajax 的所有运作原理。

1. ad hoc coding styles，指随手编写一些不具有重用性、仅仅用来解决当前问题的代码。——译者注

第6章~第8章建立在已经获得的基础知识之上,考察如何进一步完善 Ajax 应用,超越概念验证的阶段,建造一个可实际使用的、有趣的、安全的 Ajax 应用。第6章着眼于用户体验,深入考察当执行异步任务时,通知用户的方法。在妨碍用户的操作和让用户完全不了解发生了什么之间需要达到一种平衡,我们还为你展示了如何找到愉快的中间地带。

第7章从很多角度考察了 Ajax 中的安全问题。Ajax 是一种 Web 技术,它所面对的很多问题和任何其他 Web 应用没有什么不同。我们涵盖了基础的知识,在这里集中在特定于 Ajax 的问题上,例如安全地导入服务器端生成的 JavaScript,保护 Web 服务入口点免受不必要的直接操作。安全性可能是重大应用中最受关注的地方,我们提供了将安全性保持在可控范围之内所必须采取的基本步骤。

第8章讨论了其他备受关注的地方:性能(更恰当地说,是性能问题!)。我们展示了如何监视应用的性能;以及如何分析代码,以便改善性能并使这些改善在整个应用中都有效。

在第四部分(第9章~第13章),我们将转而考察很多 Ajax 项目。在每个案例中,我们都使用一种直接的方式来为要实现的功能编写代码,然后将其重构为只需几行代码就能应用在你自己项目中的健壮代码。这样做所带来的好处是可以让你更好地理解 Ajax 的原则和重用的好处,同时也为你生动地展示了如何对 Ajax 应用进行重构。

在第9章中,我们将考察通过使用 Ajax 增强 HTML 表单,带给用户更加丰富的体验的一种简单方法:使用在一个字段中输入的数据,通过以后台方式向服务器请求数据,来预置第二个下拉列表。我们在第10章中继续考察增强表单这个主题,实现一个录入前提示(type-ahead suggest),从服务器获取数据来响应用户的按键。

第11章继续探讨 Ajax 用户界面的更加广阔的可能性。我们开发了一个完整的门户(portal)应用,相比 Web 页面来说它更加类似于一个工作站桌面,使用它自己的可拖拽、可调整大小的窗口来实现。Ajax 进程以后台方式跟踪窗口的移动,所以桌面总是保持在与你离开时相同的状态,甚至当你在另外一台机器上重新登录时也是这样。

第12章开发了一个基于 Ajax 的搜索系统,演示了如何使用客户端 XSLT 将原始的 XML 数据转化为格式化的、带有样式的内容。

在第13章中,我们展示了没有后端¹实现的一个 Ajax 客户端。它仍然和服务器进程进行通信,但是在这情况下,它是使用因特网标准的 RSS 协议,直接获取服务器上的博客和新闻数据。

最后,我们给出了三个附录,希望你发现它们很有用。本书的主体讨论了 Ajax 技术本身。对于一种新的跨学科技术来说,组合各种工具来有效地使用它,相对于使用 J2EE 或 .NET 这样成熟的技术来说,更加具有挑战性。虽然厂商还没有开始提供 Ajax 开发工具,但是我们确信他们

1. 即服务器端。——译者注

一定会提供的¹！在附录 A 中概述了许多开发工具和技巧，我们正是使用它们开发 Ajax 项目，并且保持代码井然有序的。

附录 B 是为企业程序员而写的，他们理解软件设计原则，但并不是很确定如何将这些原则应用于 JavaScript 这样灵活、无组织和有些古怪的语言之上。我们探讨这种语言能够做些什么，并且指出它与 Java 和 C# 主要的分歧所在。

如果说工具厂商还没有完全跟上 Ajax 前进的脚步，框架开发者也没有跟上来。Ajax 框架领域是一个创新的和兴趣盎然（并且经常是重新发明轮子）的温床。附录 C 汇集了目前我们所知道的 Ajax 框架和工具包，并且为每一个条目提供了简短的综述和链接。

代码约定

所有的代码清单或正文中的源代码都使用一种等宽字体来与普通文字相区别。我们在本书中使用了很多种语言和标记——JavaScript、HTML、CSS、XML、Java、C#、Visual Basic.Net 和 PHP——但是尽量采用一致的方式。正文中的方法和函数名称、对象属性、XML 元素和属性都使用这种相同的字体来显示。

在很多情况下，对最初的源代码重新做了格式化。为了适应书中可用的页面空间，我们增加了换行并且重新做了缩进。在少数情况下这样做甚至还不够，代码清单还包括了续行的记号。另外，还从代码清单中去掉了许多注释。在适当的地方，我们也去掉了无助于讲述而只会使读者分心的实现细节，例如 JavaBean 的设置和获取方法、import 和 include 语句、namespace 声明。

很多代码清单都有代码注解，并且对重要的概念加以突出显示。在一些情况下，代码清单之后的注解还加上了数字标注的链接。

代码下载

本书的所有实例的源代码都可以从图灵网站下载。

我们意识到并非所有的读者都会在桌面电脑上安装 .NET 服务器、J2EE 应用服务器以及 Linux、Apache、MySQL、PHP/Python/Perl(LAMP)，而且你对本书的主要兴趣也还是在于客户端的技术。所以，我们尽量包括了基于“mock”的版本例子代码，它能和来自任何 Web 服务器(包括 Apache、Tomcat 和 IIS)的静态虚拟数据一起运行²。这些例子是在完整的工作例子之外提供的，所以如果你确实喜欢使用数据库和应用服务器，你也可以深入地研究那些完整的例子。一些基本

-
1. 正如作者预测的，2006 年 2 月初，由 IBM 倡导，Google、雅虎、Oracle 等 IT 巨头宣布联手支持一个名为“Open Ajax”的开源计划，目的是为了解决目前 Ajax 开发中面临的软件开发过程无规范以及缺乏开发和实施工具的问题。——译者注
 2. mock 是一种对象类型，在自动的单元测试中被大量使用，用于模拟外界与之通信的对象的行为。这里是指采用客户端的 mock 对象来模拟 Web 服务器的行为。——译者注

的安装文档随下载的代码包一同提供。

作者在线

购买 *Ajax in Action* 一书也意味着你获得了免费访问 Manning 出版社所运营的私有 Web 论坛的权利，在这个论坛上你可以对本书作出评论，可以提出技术问题并从作者和其他读者那里获得帮助。为了要访问论坛并且订阅其内容，可以将 Web 浏览器指向 <http://www.manning.com/crane>。该网页上提供了注册以后如何访问论坛的信息，也说明了你可以获得什么类型的帮助，并且说明了论坛的行为准则。

Manning 出版社对读者的承诺是提供一个读者之间以及读者和作者之间能够展开深入对话的场所，但是并不承诺作者必须要参与到何种程度，作者对于论坛的贡献是自愿的（并且是无偿的）。我们建议你向作者多提一些有挑战性的问题，以免他们丧失兴趣！

只要本书还在印刷销售，作者在线论坛和以前讨论的存档都可以从出版社的网站上访问到。

关于书名

通过组合介绍、综述和 how-to 的例子，实战 (*In Action*) 系列设计为有助于学习和记忆的丛书。根据认知科学的研究，人们最容易记住的事情就是他们在自发探索中发现的事情。

尽管在 Manning 出版社并没有认知科学家，我们仍然确信为了使学习变得持久，必须经历探索、实践和（很有趣的）复述学到的内容这几个阶段。只有在对新的事物进行了积极的探索之后，人们才能够理解并且记住新的事物，也就是所谓的掌握了新的事物。人类在实践中学习，例子驱动是所有 *In Action* 丛书的基本部分。这个特点鼓励读者去尝试新的事物，实践新的代码，探索新的思想。

关于书名，这里还有一个更加世俗化的理由：我们的读者都很忙。他们由于工作需要或者为了解决一个问题而寻求图书的帮助。他们要求图书允许他们很容易地读进去，又能够很容易地走出来，并且仅仅当他们在需要学习的时候，才去学习书中他们真正需要的东西。他们需要能够帮助他们增强实战能力的书。这个系列丛书就是为这样的读者而设计的。

封面图片

本书的封面图片是“Sultana”，一个苏丹家族的女性成员，苏丹的妻子和他的母亲都有可能叫这个名字。插图来自于一本土耳其奥斯曼帝国服饰集锦的画册，由伦敦老邦德街的 William Miller 于 1802 年 1 月 1 日出版。画册中的扉页已经丢失了，因此我们无法确定确切的创作时间。书中的目录使用英语和法语来标识图画，每张画有创作它的两位艺术家的名字，他们无疑会为他们的艺术能够美化两百年之后的一本计算机编程书籍的封面而感到惊讶。

Manning 出版社的一位编辑在位于曼哈顿西 26 街“Garage”的古董跳蚤市场购买了这本画

册。卖主是住在土耳其安卡拉的一个美国人，交易的时间发生在那天他准备收摊的时候。Manning 出版社的编辑随身没有携带购买这本画册所需的足够现金，并且卖主礼貌地拒绝了他使用信用卡和支票。因为卖主当天晚上要飞回安卡拉，所以似乎已经没有希望了。那么是怎么解决的呢？无非就是一个通过握手来约定的老式的君子协议。卖主提议通过银行转账付款，编辑仅仅在纸片上抄下了收款银行的信息，随后画册就到了他的手里。不用说，第二天就将款付给了卖主。我们感谢这位陌生人能够如此信任我们的同事。这使得我们回忆起了那个很久以前的美好时代。

来自奥斯曼帝国画册的图片，就像在丛书封面上出现的其他插图一样，将两个世纪之前服饰习俗的丰富性和多样性带到了我们的身边。这些图片令我们回忆起与那个时期（以及除了通信极其发达的今天之外的每一个其他历史时期）之间的隔绝和距离。

从那以后，服饰习俗所代表的不同区域的差异已经改变了，那个时期丰富多彩的服饰文化也逐渐褪色了。现在甚至连区分一个大陆的居民和另一个大陆的居民常常都是困难的。也许我们应该尽量乐观地看待这些变化，文化上、视觉上的差异已经转变为更加多样化的个人生活上的差异，或者更加多样化和更加有趣的智力上和技术生活上的差异。

我们使用来自这本画册中的图片作为本书的封面，这为我们带来了两个世纪之前丰富的、多样性的区域生活。我们 Manning 出版社的编辑们使用这种方式来赞美计算机行业中的独创性和主动性，当然，还包括其中的乐趣。