

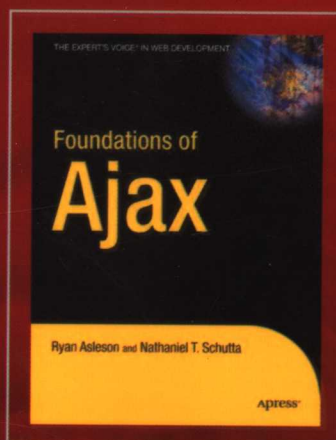
Foundations of **Ajax**

亚马逊计算机
榜首图书!

Ajax 基础教程

[美] Ryan Asleson
Nathaniel T. Schutta 著
金灵 等译

- 涵盖 Web 2.0 核心技术
- 创建桌面式 Web 应用
- 快速上手，立即给网站带来全新用户体验



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

Ajax 基础教程

Foundations of Ajax

[美] Ryan Asleson 著
Nathaniel T. Schutta

金灵 等译



人民邮电出版社
POSTS & TELECOM PRESS

图书在版编目 (CIP) 数据

Ajax 基础教程 / (美) 阿斯利森, (美) 舒塔著; 金灵等译.

—北京: 人民邮电出版社, 2006.2

(图灵程序设计丛书)

ISBN 7-115-14481-8

I. A... II. ①阿...②舒...③金... III. 计算机网络—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字 (2006) 第 003735 号

内 容 提 要

Ajax 技术可以提供高度交互的 Web 应用, 给予用户更丰富的页面浏览体验。本书重点介绍 Ajax 及相关的工具和技术, 主要内容包括 XMLHttpRequest 对象及其属性和方法、发送请求和处理响应、构建完备的 Ajax 开发工具、使用 JsUnit 测试 JavaScript、分析 JavaScript 调试工具和技术, 以及 Ajax 开发模式和框架等。本书中所有例子的代码都可以从 Apress 网站本书主页的源代码 (Source Code) 免费得到。

本书适合各层次 Web 应用开发人员和网页设计人员阅读。

图灵程序设计丛书

Ajax 基础教程

-
- ◆ 著 [美] Ryan Asleson Nathaniel T.Schutta
 - ◆ 译 金 灵 等
 - 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: 17
字数: 356 千字 2006 年 2 月第 1 版
印数: 1—5 000 册 2006 年 2 月北京第 1 次印刷

著作权合同登记号 图字: 01-2006-0319 号

ISBN 7-115-14481-8/TP · 5211

定价: 35.00 元

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

版 权 声 明

Original English language edition published by Apress L.P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA.

Copyright © 2005 by Apress L.P. Simplified Chinese language edition copyright © 2005 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。

译 者 序

见过 Google Maps 的界面吗？也许你还不以为然，如果是桌面应用，这只是雕虫小技而已。不过要知道，你看到的可不是桌面应用，它完全是一个 Web 应用！

也许你又会说，是 Web 应用又怎么样？我一样可以做到。不错，不是有这么一句话吗？“只要功夫深，铁杵也能磨成针。”只要你肯想办法，没有做不到的。但是，可以告诉你，这个应用并没有让开发人员费多大力气。

是什么“高级技术”让开发人员如虎添翼？又是什么“宝贝”使用户欣喜万分？

这就是 Ajax！

Ajax 并不是一个高深的技术，如果说是宝贝，也只能算是“老宝贝”！曾经有人评价 Ajax 是“新瓶装老酒”。但是这并不妨碍人们对它趋之若鹜。而且 Ajax 这瓶老酒经过时间的积淀和环境的变化，已经更加醇香，再加上美观实用的新包装，它的人气这么旺，可以说是顺理成章。

实际上我们从界面上看到的只是一部分，Ajax 带给我们的远不只是这些。到 Google Maps、Google Suggest 和 Gmail 上亲身感受一下，你会有更深体会。

你是不是有这样一些问题：

- Ajax 是什么？新名词？杂烩汤？还是……
- 用 Ajax 到底能做什么？制作漂亮界面？改善用户体验？还有……
- 怎么实现 Ajax？XMLHttpRequest 对象是什么？怎么避免页面完全刷新？……
- Ajax 最适用哪些情况？完成验证？建立自动刷新页面？还有……
- Ajax 有没有帮手？可以用哪些工具和技术简化 Ajax 开发？
- 能在 Ajax 开发应用测试驱动开发吗？怎么做呢？
- JavaScript 也能轻松地调试吗？有哪些调试工具？这些工具好用吗？

- 有哪些可用的 Ajax 框架？Ajax 框架对 Ajax 开发有多大作用？

本书能够帮助你回答这些问题。相信通过阅读本书，你能轻松地迈进 Ajax 殿堂，让你的老板、用户对你刮目相看！

我们深深地感谢我们的家人和朋友。在翻译过程中，他们给予了我们莫大的关心、支持和帮助。

全书由金灵主译，苏金国、徐阳、刘鑫、蔡洪亮等检查术语，刘晓兵、荆涛、张野、任岗、伊瑞海等提供技术问题支持，在大家的努力下共同完成了本书的翻译工作。

由于时间仓促，且译者的水平有限，译文中难免会出现一些错误，请读者批评指正。

译者

2005.12

前 言

几年前开始构建 Web 应用时，我们感觉这简直就是软件开发的“圣杯”。以前，我们一直开发的都是胖客户应用，公司每次发布这种公司应用的新版本时，总是需要将这种应用部署到分散在全国各地的数百个用户那里去，让我们沮丧的是，这种复杂的安装过程不仅冗长而且很容易出错，不仅让开发人员很头疼，用户也非常不满。

通过浏览器来部署应用，这看上去相当不错，因为这样，就不再需要在客户端上安装软件了。所以，与许多其他公司一样，我们公司也很快转型，开始在 Web 上部署应用。

尽管部署起来相对容易，但 Web 应用也有自己的问题。在用户看来，最突出的问题是用户界面没有了以往丰富的交互性。Web 应用仅限于使用 HTML 提供的一组基本部件，而这是很有限的。更糟糕的是，与服务器交互需要完全刷新页面，很多用户已经熟悉了功能强大的客户-服务器应用，对他们来说，这一点很让人不快。

我们曾经一直认为，在 Web 应用中只要刷新页面就必须完全刷新，好像这是在所难免的，所以往往想方设法地避免页面刷新。我们甚至还考虑过编写一个 Java applet，由它处理浏览器和服务器之间的通信。不过，随着越来越多 Web 应用的部署，我们很快发现，用户已经习惯了这种完全页面刷新的方式，这么一来，我们也不再那么强烈地想要另辟蹊径了。

转眼 5 年过去了。由于 Google Suggest 和 Gmail 等应用的出现，甚至在 Ajax 这个术语出现之前，这种使用 XMLHttpRequest 对象在浏览器和服务器之间完成异步通信的方法就已经在开发者社区中产生了很大反响。多年之前，IE 中就已经使用了 XMLHttpRequest 对象，但是如今它得到了更多其他浏览器的支持，取得了重大突破。我们在一个正在开发的应用中增加了 Ajax 功能，结果令我们震撼不已，所以我们都产生了一种想法：“嘿，应该有人来写一本有关 Ajax 的书呀。”本书因此应运而生。

本书概述

本书旨在为开发人员介绍为已有的或者将来的新应用增加 Ajax 技术所需的所有工具。

在写作中我们牢记：“你需要知道的我们都要介绍，你不需要知道的我们绝口不提。”我们认为，作为本书的读者，你应该已经是一个有经验的 Web 应用开发人员。正因如此，我们会把重点放在你很可能不了解的新内容上：Ajax 及相关的工具和技术。我们不会花大量篇幅来讲述服务器端语言，因为我们认为你会自己选择工具集来开发服务器端功能，在这方面你不需要我们的帮助。另外我们也不会浪费时间来讨论如何构建企业级应用，这样的应用很少使用 Ajax。相反，重申一次，我们的重点只是 Ajax 以及相关的工具和技术。

本书中的示例特意做得很小，而且很紧凑。它们会尽可能简洁地展示一个或两个重要的 Ajax 概念。我们认为，作为一个有经验的 Web 开发人员，你应该能熟练地把我们展示的内容推广到自己的环境中去，因此，我们不会在示例中“堆放”对你没有多大用处的信息。

第 1 章讨论了 Web 应用开发的发展历程，从过去谈到现在，并且预测了将来。如果你了解了 Web 开发技术的过去，就能更容易地认识到它们将来会有怎样的发展。

第 2 章介绍了 XMLHttpRequest 对象。这个 Ajax 概念你可能不太熟悉，所以我们专门用一章来解释 XMLHttpRequest 对象的属性和方法。也许你像我们一样，直到最近才注意到 XMLHttpRequest 对象。不过，要知道早在几年前 IE 中就已经有了 XMLHttpRequest 对象。因此，我们会用充分的笔墨讨论 XMLHttpRequest 对象，并说明它能做什么。

第 3 章开始讲述 Ajax 的具体内容。这一章将讨论 XMLHttpRequest 对象用来与服务器通信的各种方法。我们讨论了可以使用 XML、纯文本甚至 JavaScript Object Notation (JSON) 作为传输介质，并介绍了它们与 XMLHttpRequest 对象结合使用的各种方式。在这一章的最后，你就能熟练地使用 XMLHttpRequest 对象与服务器端通信，而不必让用户苦苦等待页面完全刷新了。

作为开发人员，我们总是花很多时间学习新技术，但并不知道怎么实际应用。第 4 章将针对这个问题展示一系列可以使用 Ajax 技术的场合。如前所述，每个示例都很小、很紧凑，这样你能更好地理解相应内容，而不用在大堆不必要的信息中搜寻。

第 5、6 和 7 章对于 Ajax 新手来说尤其重要。我们不希望你仓促上阵，应该先配备好合适的工具和技术，再考虑在应用中增加 Ajax。利用第 5 章介绍的工具和技术，可以简化 Web 应用的开发，你可以得到高质量、遵循行业标准而且将来更易于维护的代码。

测试驱动开发 (Test-driven development, TDD) 使我们开发应用的方法发生了根本性的改变。在编写代码之前先编写单元测试，这样就能确保所写的代码会正常工作，并能大大提高代码的质量。利用一个单元测试集，可以确保做出修改后所有代码还能按预期的那样工作，这样将来进行修改将更容易。Ajax 当然也应该使用 TDD，而且 TDD 的好处再强调也不为过，所以我们专门用一章来讨论。由于 Ajax 主要是一个基于浏览器的技术，第 6 章将展示如何对 JavaScript 代码应用 TDD。

谈到 JavaScript, 如果你想使用 Ajax, 就必须至少用 JavaScript 写过一些程序。许多开发人员都不喜欢 JavaScript, 认为它缺少一些重要的生产性工具, 如调试器, 所以没有多大的用处。如今这种说法已经不成立了。第 7 章讨论了一些调试工具和技术, 如果出现问题, 你可以用这些工具和技术来跟踪, 使得问题可以尽快、尽可能容易地加以解决。你不用再担心在使用 JavaScript 时出现问题而不能诊断, 也不用因此对 JavaScript 退避三舍。

Ajax 是一个发展迅猛的技术, 在写本书期间, Ajax 已经得到了飞速的发展。第 8 章把所有的内容综合在一起, 讨论了新兴的 Ajax 开发模式和框架, 并提供了一些在线资源。另外, 第 8 章还给出了一个完整的示例, 不仅显示了一些高级的 Ajax 技术, 还展示了使用现成的 Ajax 框架来完成 Ajax 开发是何等容易。通过使用框架, 你不用再做 Ajax 开发的一些繁琐任务, 这样就能把重点集中在业务逻辑上, 而不是 Ajax 的细节上。

最后, 附录 A 描述了不同浏览器上 W3C DOM 和 JavaScript 实现中存在的一些特异之处和不一致的地方, 并提供了一些方法来解决这些问题。附录 B 总结了最流行的 Ajax 框架和库, 这些框架和库有助于简化 Ajax 技术的使用。随着 Ajax 越来越流行, 框架也肯定会越来越多, 所以让我们对新的框架和其他开发工具拭目以待。

本书源代码

本书中的所有示例都可以从 Apress 网站的源代码 (Source Code) 部分免费得到。在浏览器地址栏中输入 www.apress.com, 点击 Source Code 链接, 在列表中找到 *Foundations of Ajax*, 也可以从 Apress 网站的本书主页下载源代码 (zip 文件)。源代码是按章组织的。

本书的更新信息

尽管我们尽了最大努力, 但是你可能还会在书中偶尔发现一些错误, 当然我们希望不会这样! 如果正文或者源代码中还有错误, 对此我们表示歉意。可以从 Apress 网站的本书主页上得到最新的勘误表, 在那里还会提供我们的联系信息, 如果你发现了错误, 可以通知我们。

联系我们

我们非常重视你对本书内容和源代码示例的疑问和建议。请把所有问题和建议直接发到 foundationsofajax@gmail.com, 我们会尽快回复。不过要记住, 我们 (像你一样!) 不可能马上就有回应。

谢谢你购买了这本书! 我们希望你认为这是一本有价值的书, 能像我们写书一样兴趣盎然地阅读这本书。

Ryan Asleson 和 Nathaniel T. Schutta

致 谢

感谢 Apress 提供了编写本书的机会。感谢 Grace Wong 在整个写作过程中一直督促着我们集中精力、保证进度。Keith Harvey 和 Brent Ashley 提供了很有价值的反馈意见，本书能写成这样，离不开他们的帮助。Kim Wimpsett 很是倒霉，他要对付我们成堆的拼写和语法错误，对此我们致以衷心的感谢。我们要感谢 Linda Marousek 指导我们完成了最后的出版过程，看到用电脑打印的手稿变成一本装帧精美的图书，实在让我们欣喜万分。Ewan Buckingham 回答了我们很多问题，让我们能轻松地进入写作过程。还要感谢在整个过程中我们的代理人 Laura Lewin 和 Studio B 的全体人员所提供的支持。

—Ryan Asleson 和 Nathaniel T. Schutta

我要感谢整个 Apress 团队，能让我这样一个没有写作经验的新手写这本书。没有他们的信任和支持，我是无法完成这本书的。我要感谢 Gary Cornell，在我最早提议写这本书后，他给我回了一封热情洋溢、充满信任和鼓励的电子邮件。我要特别感谢 Chris Mills，是他帮助我反复修改提纲，终于有了现在这个最终稿。

对于 Nate Schutta，我的朋友、同事和合作者，说再多好话也不为过。我实在钦佩他旺盛的精力、卓越的见识和为这本书付出的辛勤努力。没有他的帮助，这本书就不可能出版。

最重要的，我要感谢妻子 Sara，在我忙于写书的几个月里，她独自承担了很多责任，但却毫无怨言。

—Ryan Asleson

感谢我的合作者 Ryan，谢谢你让我帮忙，把你的想法变成一本书，这真是我的荣幸。那一天在会议中间休息的时候，我只是向你扔过去一个球，谁想到会带来这样一个结果呢？说到这里，我还要感谢 Sara 和 Adam！当然，要是没有感谢我可爱的妻子 Christine，

实在是说不过去。在写书的期间，你如此宽容，如果没有你的爱和耐心，我什么也做不到。非常感谢 Nathan Good，感谢你的帮助和建议，否则我不会有今天的成绩（我会怀念与你共饮咖啡的那些时光）。

没有我父母的远见，我可能不会涉足这个领域。多年前的一个夏天，他们给我报名参加了一个暑期计算机学习班，而且给我用了当时最新潮的机器。谢谢你们，对你们的感谢无以言表！另外，还要特别感谢圣约翰大学计算机科学系的 Jim Schnepf、Lynn Ziegler、John Miller、Andy Holey 和 Noreen Herzfeld，你们真是好心，能接受我这样一个化学专业的学生攻读计算机专业的硕士，我才有了这样一个非比寻常的学习历程！我还要感谢很多对我有影响的老师：Karen Sweet、Mary Ellen Briel、Michael Youngberg、Peggy Anderson、Jim Murphy（已故），当然还有 Elizabeth Stoltz。谢谢你们大家，你们让我的生活更加丰富，对此我无以为报。没有你们的指导和监督，我可能不是今天的我。我知道，由于篇幅所限，我肯定还遗漏了很多，对此我致以深深的歉意。再次感谢这里提到的每一个人，还有那些永远铭刻在我心里的人们。

——Nathaniel T. Schutta

关于技术审校

■ KEITH HARVEY 是 SCOPE iT 公司 (www.scopeit.com) 的 CTO 和首席架构师。该公司主要致力于开发基于 Web 的项目预算应用程序,用于 IT 项目的快速估算、预算和规划。SCOPE iT 是 Microsoft 的合作伙伴,拥有 ISV/软件解决方案资格。SCOPE iT 应用程序是在最新的 Microsoft .NET 技术、SQL Server 和 Ajax 的基础上开发的。

Keith 也是一位作者,写过许多关于 Microsoft 技术、数据库、Ajax、软件项目估算等方面的文章。Keith 与妻子 Tricia 和女儿 Hanna 居住在北加州。他的个人网站是 www.keith-harvey.com。

目 录

译者序

前言

致谢

关于技术审校

第 1 章 Ajax 简介	1
1.1 Web 应用简史	1
1.2 浏览器历史	2
1.3 Web 应用的发展历程	3
1.3.1 CGI	4
1.3.2 applet	4
1.3.3 JavaScript	5
1.3.4 servlet、ASP 和 PHP.....哦，太多了！	6
1.3.5 Flash	9
1.3.6 DHTML 革命	10
1.3.7 XML 衍生语言	10
1.3.8 基本问题	12
1.3.9 Ajax	13
1.4 可用性问题	17
1.5 相关技术	18
1.6 使用场合	19
1.7 设计考虑	20
1.8 小结	21

第 2 章 使用 XMLHttpRequest 对象	23
2.1 XMLHttpRequest 对象概述	23
2.2 方法和属性	25
2.3 交互示例	26
2.4 GET 与 POST	28
2.5 远程脚本	29
2.5.1 远程脚本概述	29
2.5.2 远程脚本的示例	29
2.6 如何发送简单请求	31
2.6.1 简单请求的示例	32
2.6.2 关于安全	34
2.7 DOM Level 3 加载和保存规约	35
2.8 DOM	35
2.9 小结	36
第 3 章 与服务器通信：发送请求和处理响应	37
3.1 处理服务器响应	37
3.1.1 使用 innerHTML 属性创建动态内容	37
3.1.2 将响应解析为 XML	40
3.1.3 使用 W3C DOM 动态编辑页面	45
3.2 发送请求参数	52
3.2.1 请求参数作为 XML 发送	59
3.2.2 使用 JSON 向服务器发送数据	64
3.3 小结	70
第 4 章 实现基本 Ajax 技术	71
4.1 完成验证	71
4.2 读取响应首部	75
4.3 动态加载列表框	79
4.4 创建自动刷新页面	85
4.5 显示进度条	90
4.6 创建工具提示	95
4.7 动态更新 Web 页面	101
4.8 访问 Web 服务	110

4.9 提供自动完成	116
4.10 小结	123
第 5 章 构建完备的 Ajax 开发工具箱	125
5.1 使用 JSDoc 建立 JavaScript 代码的文档	125
5.1.1 安装	126
5.1.2 用法	126
5.2 使用 Firefox 扩展验证 HTML 内容	129
5.2.1 HTML Validator	130
5.2.2 Checky	132
5.3 使用 DOM Inspector 搜索节点	133
5.4 使用 JSLint 完成 JavaScript 语法检查	137
5.5 完成 JavaScript 压缩和模糊处理	138
5.6 使用 Firefox 的 Web 开发扩展	140
5.7 实现高级 JavaScript 技术	141
5.7.1 通过 prototype 属性建立面向对象的 JavaScript	142
5.7.2 私有属性和使用 JavaScript 的信息隐藏	146
5.7.3 JavaScript 中基于类的继承	148
5.7.4 汇合	149
5.8 小结	152
第 6 章 使用 JsUnit 测试 JavaScript 代码	155
6.1 JavaScript 提出的问题	155
6.1.1 测试先行方法介绍	155
6.1.2 JUnit 介绍	157
6.2 分析 JsUnit	158
6.2.1 起步	159
6.2.2 编写测试	159
6.2.3 运行测试	172
6.2.4 使用标准/定制查询串	177
6.2.5 使用 JsUnit 服务器	181
6.2.6 获得帮助	183
6.2.7 还能用什么?	184
6.3 小结	184

第 7 章 分析 JavaScript 调试工具和技术	185
7.1 用 Greasemonkey 调试 Ajax 请求	186
7.1.1 Greasemonkey 介绍	186
7.1.2 使用 Greasemonkey XMLHttpRequest 调试用户脚本	186
7.1.3 使用 XMLHttpRequest 调试用户脚本检查 Ajax 请求和响应	186
7.2 调试 JavaScript	188
7.2.1 使用 Firefox JavaScript Console	189
7.2.2 使用 Microsoft Script Debugger	190
7.2.3 使用 Venkman	192
7.3 小结	207
第 8 章 万事俱备	209
8.1 模式介绍	209
8.1.1 实现褪色技术	209
8.1.2 实现自动刷新	210
8.1.3 实现部分页面绘制	210
8.1.4 实现可拖放 DOM	211
8.2 避免常见的陷阱	212
8.3 相关的更多资源	214
8.4 使用框架	216
8.5 Taconite 介绍	216
8.5.1 Taconite 原理	217
8.5.2 解决方案	217
8.5.3 Taconite 怎么处理内容	219
8.6 Dashboard 应用介绍	219
8.7 用 Taconite 构建 Ajax Dashboard	221
8.7.1 一般特性介绍	221
8.7.2 设计特性介绍	222
8.7.3 分析代码	224
8.7.4 分析天气预报组件	225
8.7.5 分析标题新闻组件	232
8.7.6 如何完成自动重新刷新工作	235
8.7.7 构建更好的 autocomplete	237
8.8 小结	240

附录 A 开发跨浏览器 JavaScript	241
A.1 向表中追加行	241
A.2 通过 JavaScript 设置元素的样式	242
A.3 设置元素的 class 属性	243
A.4 创建输入元素	243
A.5 向输入元素增加事件处理程序	244
A.6 创建单选钮	245
A.7 小结	246
附录 B Ajax 框架介绍	247
B.1 浏览器端框架	247
B.1.1 Dojo	248
B.1.2 Rico	248
B.1.3 qooxdoo	248
B.1.4 TIBET	249
B.1.5 Flash/JavaScript 集成包	249
B.1.6 Google AJAXSLT	249
B.1.7 libXmlRequest	249
B.1.8 RSLite	250
B.1.9 SACK	250
B.1.10 sarrisa	250
B.1.11 XHConn	251
B.2 服务器端框架	251
B.2.1 CPAINT	251
B.2.2 Sajax	251
B.2.3 JSON/JSON-RPC	251
B.2.4 Direct Web Remoting	252
B.2.5 SWATO	252
B.2.6 Java BluePrints	252
B.2.7 Ajax.Net	252
B.2.8 Microsoft 的 Atlas 项目	253
B.2.9 Ruby on Rails	253