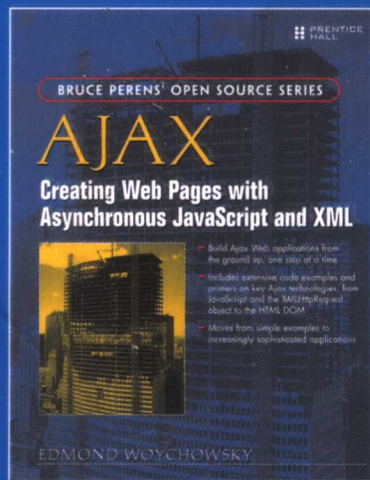


From Programmer, For Programmer

- 从零开始一步步创建Ajax Web 应用程序
- 包含大量示例代码以及Ajax 关键技术, 涵盖JavaScript、XMLHttpRequest对象及HTML DOM
- 从简单的实例逐渐深入到复杂的应用程序



Ajax

关键技术 与典型案例

[美] Edmond Woychowsky 著
姚军 徐锋 陈绍继 译

Ajax 编程入门

- 1.1 Ajax 概述
- 1.2 Ajax 的组成
- 1.3 Ajax 的应用



Ajax 入门

第 1 章 绪论

From Programmer,
For Programmer

Ajax

关键技术 与典型案例

〔美〕Edmond Woychowsky 著
姚军 徐锋 陈绍继 译

人民邮电出版社

北京

图书在版编目 (CIP) 数据

Ajax 关键技术与典型案例/ (美) 沃伊乔斯基 (Woychowsky, E.) 著; 姚军, 徐锋, 陈绍继译. —北京: 人民邮电出版社, 2007.9

ISBN 978-7-115-16524-4

I. A... II. ①沃...②姚...③徐...④陈... 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 (2007) 第 103070 号

版 权 声 明

Authorized translation from the English language edition, entitled Ajax: Creating Web Pages with Asynchronous JavaScript and XML, 0132272679 by Edmond Woychowsky, published by Pearson Education, Inc, publishing as Prentice Hall, Copyright © 2007 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 & TELECOMMUNICATIONS PRESS Copyright © 2007.

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。

Ajax 关键技术与典型案例

- ◆ 著 [美] Edmond Woychowsky
译 姚 军 徐 锋 陈绍继
责任编辑 刘映欣
- ◆ 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>
三河市海波印务有限公司印刷
新华书店总店北京发行所经销
- ◆ 开本: 800×1000 1/16
印张: 24.5
字数: 517 千字
印数: 1-4 000 册

2007 年 9 月第 1 版

2007 年 9 月河北第 1 次印刷

著作权合同登记号 图字: 01-2006-7581 号

ISBN 978-7-115-16524-4/TP

定价: 55.00 元

读者服务热线: (010) 67132705 印装质量热线: (010) 67129223



内容提要

本书是为各个层次的 Web 开发人员准备的 Ajax 教程，它不仅系统地介绍了 Ajax 及其相关的 XML 和 XMLHttpRequest 技术，还阐述了 HTML 和 JavaScript 等基本技术，以及 XSLT 和 XPath 等较高深的 Web 技术。本书通过一个个小型的、可以正常工作的、逐步演化的实例，以“润物细无声”的方式使读者理解其基本原理，掌握应用技巧。

本书对于希望了解 Ajax、开发出更好的 Web 应用程序的开发人员是一本很好的参考手册，同时它也是 Ajax 应用开发人员的速查手册。书中所有源代码可在网址 <http://www.ptpress.com.cn> 的下载区免费下载。



作者简介

Edmond Woychowsky 毕业于 Middlesex 郡学院和宾夕法尼亚州立大学，在贝尔实验室开始他的职业生涯，在那里他负责编写用于 DOSS 订单输入系统的恐龙级的递归式汇编语言程序。在他的职业生涯中，他曾经在银行、保险、制药和制造等多种行业工作，逐渐羽翼丰满并成长为一名 Web 开发人员。他因经常在 TechRepublic 网站上发表拥有独到见解的论文，而且能够简明扼要地介绍 Muenchian 分组的工作机制而闻名。目前，他在新泽西担任顾问工作，经常以奇妙的方式应用 Ajax 和 XSLT 技术解决问题，这已成为他下一个“赖以谋生”的技术。



前言

本书意在简明地阐释 Ajax 应用程序开发的基础知识。

关于本书

近几年，在 Web 应用程序开发领域潜移默化地经历着一场革命。这场革命是如此平静，以至于直到 2005 年 2 月之前，都没有人为这场革命命名，包括革命者本身。确实，除了不断提起那些诸如 XMLHttpRequest 对象、XML 及 SOAP 之类的奇怪词语，开发者们并没有怎么提到这场革命，可能是害怕因挑战常规而遭受火刑。但是现在这已经不是什么秘密了，没有任何理由不让人们知道 Ajax 的工作原理。

因为我是一个“在实践中学习”的信奉者（无需 Kool Aid¹），所以在本书中能够发现更多代码实例。本书就是为那些喜欢实验更胜于听人说教的读者所编写的。如果“喜欢”这个词不恰当，也可以用诸如“从……学到更多”之类的词来代替。

直到 2005 年左右，信奉“在实践中学习”的开发人员都不太相信一张叫做认证证书的纸能够比真实案例中获得的经验更重要。我想在某种程度上，这种想法没有问题。但是不幸的是，当工作机会变得越来越少时，开发人员就开始像马科斯夫人收藏鞋子一样地收藏各种证书。百科全书式的知识对于面试和随后的工作可能有所帮助，但是对于保住工作机会没有太大的作用。现在钟摆开始摆向了另一个方向，其实掌握某个领域知识变得比取得它的证书更重要。这就引出了一个问题，“为什么要学习 Ajax”。

这个问题的答案可以很简单扼要，也可以像 Ajax 的概念一样丰富多彩。让我们首先从简单的回答开始，因为它看起来更适合被放在简历上。我们都知道适于放在简历上的东西是我们早已习惯的，就像生活和饮食的风格一样。如果再有一些实际动手的能力，保住工作的机会就大大增加了。

各种丰富多彩的答案多半都源于那些描述 Web 开发趋势的人们的说法，Ajax 是未来的潮流。

¹ 译者注：Kool-Aid 原意为“苦艾饮料”，是一种便宜的非碳酸软饮料，它是一种粉末，一般是用水泡并加入糖来饮用。其引申义为一种可为某种信仰或理念而壮烈牺牲的精神或行为。

当然这也会招致别人这样的回应,“以前人们也说 DHTML 是未来的潮流,可是大家不谈论它已经有 5 年之久了”。的确以前是有人说过 DHTML 是未来的潮流,但是这次的情况有所不同。

其区别在于该技术是自然发展起来的,而不是因为开发者们纷纷在简历中炫耀而形成的潮流。现在有许多实际应用例子,其效果已经远远超过了诸如跟随鼠标指针扫地的小精灵之类的小把戏了。应用这些技术的公司是真正具有更长历史的公司,利用这些技术也不再只是因为“它很酷”了。

阅读本书前应该具备的知识

本书假定读者对 Web 开发技术有基本的了解,而不仅仅是诸如所见即所得、鼠标拖放等当前标准。当然并不是必须能够手动编写 HTML 代码,只要知道 HTML 的存在就行了。本书希望能够填补一些空白,因此将涉及一些必不可少的基础知识。

除了我对拖放式的 Web 开发方法不太喜欢以外,了解 HTML 还有一个合乎逻辑的理由。那就是当 HTML 文档被装入浏览器之后,我们还将会修改它。这不是说需要对文档进行大刀阔斧地处理,只是移走元素、放入元素或者适当地修改元素。

即使不熟悉 JavaScript 也没关系,本书已经努力地来深入地阐述它,无需借助别的书籍理解本书。不过现在想一想,也失去了编写一本配套的 JavaScript 书籍的机会。

如果不熟悉 XML,也不要因 Ajax 是异步的 JavaScript 和 XML 的缩写而感到担忧,因为你所需要了解的知识这里都有。XSLT 也一样,它是一种将 XML 转换为其他格式的语言。想想霍格沃茨魔法学校,就会明白了。

本书描述了 Ajax 的演化过程(如果你喜欢,也可以说是聪明的设计),从应用动态 HTML 进行 Web 开发,直到异步的 JavaScript 和 XML。因为本书描述的是比较先进的 Web 开发技术,所以应当使用当前最好的浏览器,诸如 Firefox 或者 Flock,还需要一个 Internet 连接。

本书的结构

下面是本书各章节的概要:

- 第 1 章“网页的类型”概要地介绍了自 Web 诞生以来的各种网页编码方法,介绍了从静态网页到动态网页的 Web 开发发展史。另外还讨论了各种 Web 开发技术,最后讨论了浏览器之间的竞争。

- 第2章“Ajax简介”介绍了 Ajax 及演示第一个 Ajax 应用程序时会发生的一些情况,揭示了 Ajax 的概念,并逐步深入地介绍从原始的 Ajax 到现在的整个演变过程。

- 第3章“HTML/XHTML”描述了 Ajax 中未被提及的基本部分,包括 HTML/XHTML 和层叠样式表。

- 第4章“JavaScript”是对 JavaScript 的概述,包括数据类型、变量和操作符。还介绍了流程控制语句、递归函数、构造函数和事件处理程序等。

- 第5章“在 Ajax 中使用 HTML 和 JavaScript”描述了 Ajax 的前身。本质上,它类似于使用石刀和熊皮之类的方法。尽管这些技术有点过时,但是它能很好地展示 Ajax 应用程序是如何工作的。另外,本章还介绍了在框架中交换信息的“隐含技术”。此外,为了安抚坚信这些技术已经过时的人们,还介绍了 MySQL 中存储过程的相关知识。

- 第6章“XML”介绍了 XML,特别是 Ajax 中将应用到的那部分知识。对元素、属性和实体,各种描述内容的方法,文档类型定义,框架等都做了介绍。还包括跨浏览器的 XML 数据孤岛。

- 第7章“XMLHttpRequest”剖析了 XMLHttpRequest 对象,描述了它的各种属性和方法。对使用同步方式代替异步方式感兴趣吗?在这一章里可以找到答案。另外,本章还讨论了 Web 服务和 SOAP。

- 第8章“使用 XML 和 XMLHttpRequest 对象的 Ajax”介绍了人们所认知的纯 Ajax,特别关注了整个工作的核心——XMLHttpRequest 对象。另外,还讨论了从 PHP 到 C#各种终端系统,并介绍了 RPC 和 SOAP 这两个更流行的通信协议。

- 第9章“XPath”详细介绍了 XPath。本章从 XPath (经常被认为是 XSLT 的仆人)的基础知识开始,阐述了在一个 XML 文档中查找信息的方法。其中还阐述了 XPath 轴 (aixs) 的细节信息,至少是值得一看的。

- 第10章“XSLT”详细讲解了 XSLT 中令人称奇的主题以及如何将它应用在一个跨浏览器的 Ajax 应用当中。本章阐述了从基础知识到高级特性方面的内容,试图揭开 XSLT 神秘的面纱。

- 第11章“在 Ajax 中应用 XSLT”通过对 XSLT 的介绍进一步讲解了本书前4章所提到的素材。直到不久以前,应用 XSLT 还被认为是个坏主意,不过只要在实际应用时慎重一些,就不会有问题。XSLT 是一种可以进一步增加网站访问者体验的工具。

- 第12章“代码复用”介绍了一个在本书中所有实例都将应用到的、自己开发的客户端 JavaScript 程序库。虽然并不一定必须使用这个程序库,但是该例子能够对大部分现存 Ajax 程序库的技术内幕有较好的了解。

- 第13章“Ruby on Rails 之旅”简单介绍了开放源码的 Ruby on Rails 框架。从获取和安装各种组件开始,本章展示了启动 WEBrick Web 服务器的方法。在之后的一组例子中,演示了一个访问 MySQL 数据库的简单网页。

- 第 14 章“和 Ruby 一起走得更远”进一步深入研究了 Ruby on Rails, 并介绍了一个应用 Rails 内建 JavaScript 程序库的简单 Ajax 应用程序。
- 第 15 章“跨浏览器 HTML DOM 的本质”描述了跨浏览器的 HTML 文档对象模型的模糊及神秘的部分。作为 Ajax 未被提及的组成部分, HTML DOM 是访问 HTML 或者 XHTML 文档各个部分的重要方法。这也是 Ajax “只更新文档的一部分”特性的基础。
- 第 16 章“其他相关内容”描述了一些在万维网上可用的资源, 从可应用于 Ajax 的预编制的 JavaScript 程序库到多种可用于个人电脑的浏览器均有涉及。

本书中的约定

本书中的例子、代码片段和文本中的代码都使用代码字体。在你所选择的编辑器中按照这种风格输入, 结果可能如下所示:

```
if(enemy = 'troll')  
    runaway();
```



鸣 谢

尽管本书由我完成，但是它在许多方面都受到了许多人的影响（都是好的影响）。因为这些人在本书的创作过程中所扮演的角色非常重要，我希望在这里花一点时间来感谢我能想得起来的所有人。

感谢 Mary Ann Woychowsky，她对我沉迷于写作给予理解，当她逮到我在应该写作的时候玩“Morrowind”游戏时还会提醒我，“我猜书一定写完了吧？”感谢 Benjamin Woychowsky，每当我玩电脑游戏的时候，他也总会提醒我，“你是不是该写书了？”感谢 Crista Woychowsky，一旦被她发现我在应该写作的时候看“Star Gate SG-1”时，整套影碟就都不见了。

感谢我的母亲 Nan Gerling，她把对阅读的热爱遗传给了我并且帮助我坚持阅读。

感谢 Prentice Hall 的 Eric Garulay，他负责本书的市场工作并向我引荐了 Catherine Nolan。感谢 Prentice Hall 的 Catherine Nolan，感谢她对本书的信任和对我刚开始写作时的支持。感谢 Bruce Perens，因为我使用的是 Firefox 浏览器，在没有消除所有可能出现的问题时仍然对我保持信心。感谢 Prentice Hall 的 Denise Mickelson，他不断地提醒我写作进度。感谢 Prentice Hall 的 Chris Zahn，感谢他的编辑工作，对我那些总是很奇怪的问题的解答，以及他对事情的总体见解。感谢 George Nedeff 在编辑和印制工作环节的努力，感谢 Heather Fox 全程追踪了整个项目的进程。由于水平有限，本书仍会遗留一些问题，敬请谅解。

我还要感谢已故的 Jack Chalker 对我在写作合同中的帮助，更重要的是在整个过程中他都用我能理解的话与我交谈。还要感谢他写的许多科幻小说，这些书籍影响了我对世界的看法，毕竟，最终每一件事情都和我们对世界的看法有关。

感谢 Dossy Shiobara 为我解答了关于 MySQL 的古怪问题。

感谢 Richard Behrens 帮我更清晰地阐述自己想法。

感谢 Joan Susski，她让我确保在阐述本书用到的各种技术时保持一致的深度。

感谢 Premkumar Ekkaladevi 帮助我决定本书的技术深度。

感谢 David Sarisoehn，他多年前就向我解释了许多拒绝晦涩代码的诸多易于理解的理由。

最后还要感谢 Francis Burke、Shirley Tainow、Thomas Dunn、Marion Sackrowitz、Frances Mundock、Barbara Hershey、Beverly Simon、Paul Bhatia、Joseph Muller、Rick Good、Jane Liefert、Joan Litt、Albert Nicolai 和 Bill Ricker，是你们让我知道如何更好地学习。



目 录

第 1 章 网页的类型	1
1.1 静态网页	2
1.2 动态网页	2
1.2.1 HTML	4
1.2.2 CSS	4
1.2.3 JavaScript	5
1.3 Web 浏览器	6
1.3.1 Microsoft Internet Explorer	6
1.3.2 基于 Mozilla 的浏览器	7
1.3.3 Linux 平台下的浏览器	8
1.3.4 其他浏览器	9
1.4 跨浏览器开发简介	9
1.4.1 浏览器大战的牺牲品	10
1.4.2 市场份额不代表一切	10
1.4.3 维和部队——万维网联盟 (W3C)	11
1.5 Web 服务器端组件	11
1.5.1 Apache	11
1.5.2 Internet 信息服务器 (IIS)	12
1.5.3 其他 Web 服务器	12
1.6 在实践中学习	12
1.6.1 手动编码	13
1.6.2 制造工具的工具	13
1.7 小结	14
第 2 章 Ajax 简介	15
2.1 Ajax 不是个模型	15

2.2	Ajax 名称的由来	16
2.3	Ajax 的定义	17
2.3.1	Ajax 的价值体系	17
2.3.2	初涉非常规力量	18
2.4	与 Ajax 的第一类接触	19
2.4.1	看不见的世界	23
2.4.2	引入 JavaScript	23
2.5	与 Ajax 的第二类接触	24
2.5.1	XML	25
2.5.2	XMLHttpRequest 对象	27
2.6	与 Ajax 的第三类接触	29
2.6.1	XSLT	29
2.6.2	主旋律的变奏曲	34
2.7	新事物的形态	35
2.8	小结	35
第 3 章	HTML/XHTML	37
3.1	HTML 与 XHTML 的差异	37
3.1.1	不规范的格式	38
3.1.2	规范格式	38
3.1.3	规范格式的实例	39
3.2	元素和属性	40
3.2.1	XHTML 元素及其属性概述	40
3.2.2	隐藏帧与可见帧	54
3.2.3	运用自己的元素和属性	55
3.2.4	CSS 初步	56
3.3	小结	59
第 4 章	JavaScript	60
4.1	数据类型	61
4.1.1	数值型	61
4.1.2	字符串型	61
4.1.3	布尔型	65
4.1.4	杂项	65

4.1.5 数组	66
4.1.6 对象	67
4.2 变量	67
4.3 运算符	68
4.4 流控制语句	69
4.4.1 条件	69
4.4.2 循环	72
4.5 函数	74
4.6 递归	75
4.7 构造函数	77
4.8 事件句柄	81
4.9 小结	83
第 5 章 应用 HTML 和 JavaScript 实现 Ajax	85
5.1 隐藏帧和 iframe	86
5.2 跨浏览器的 DOM	87
5.2.1 JavaScript、ECMAScript 和 JScript	93
5.2.2 待解决的问题	99
5.3 表格式信息	102
5.3.1 只读的	107
5.3.2 可修改的	117
5.4 表单	123
5.4.1 只读的	123
5.4.2 可修改的	129
5.5 优点与缺点	138
5.6 小结	138
第 6 章 XML	139
6.1 元素	141
6.2 属性	142
6.3 禁用字符的处理	143
6.3.1 实体	144
6.3.2 CDATA 段	144
6.4 注释	145

6.5	预期处理	145
6.5.1	命名空间	146
6.5.2	DTD	146
6.5.3	模式	147
6.6	XML 声明	149
6.7	处理指令	149
6.8	XML 数据岛	150
6.8.1	Internet Explorer	150
6.8.2	Firefox	150
6.9	小结	156
第 7 章	XMLHttpRequest 对象	157
7.1	同步	158
7.2	异步	159
7.3	Microsoft Internet Explorer	161
7.4	XML 文档对象模型	162
7.5	RSS	173
7.6	Web 服务	174
7.6.1	Web 服务的定义	175
7.6.2	SOAP	177
7.7	小结	181
第 8 章	使用 XML 和 XMLHttpRequest 的 Ajax	182
8.1	传统网站与 Ajax 网站	182
8.2	XML	185
8.2.1	规范的格式	185
8.2.2	IE 中的数据岛	189
8.2.3	对所有人可用的数据岛	191
8.2.4	绑定	194
8.3	XMLHttpRequest 对象	201
8.3.1	避免卸载/重载循环	201
8.3.2	浏览器之间的不同	201
8.3.3	了解 SOAP	213
8.4	再次遇到的问题	215

8.5 表格化的信息和表单	220
8.5.1 只读的	231
8.5.2 可修改的	235
8.6 优点和缺点	237
8.7 小结	237
第 9 章 XPath	238
9.1 查找路径	240
9.2 上下文结点	241
9.3 父结点	241
9.4 属性结点	241
9.5 断言	242
9.6 XPath 函数	243
9.6.1 逻辑函数	243
9.6.2 数值函数	243
9.6.3 结点集函数	244
9.6.4 字符串函数	244
9.7 XPath 表达式	246
9.8 XPath 并操作	246
9.9 轴	246
9.9.1 Ancestor 轴示例	247
9.9.2 ancestor-or-self 轴示例	248
9.9.3 attribute 轴示例	248
9.9.4 child 轴示例	248
9.9.5 descendant 轴示例	249
9.9.6 descendant-or-self 轴示例	249
9.9.7 following 轴示例	250
9.9.8 following-sibling 轴示例	250
9.9.9 namespace 轴示例	251
9.9.10 parent 轴示例	251
9.9.11 preceding 轴示例	252
9.9.12 preceding-sibling 轴示例	252
9.9.13 self 轴示例	252
9.10 小结	253

第 10 章 XSLT	254
10.1 递归样式表与迭代样式表	255
10.1.1 范围	260
10.1.2 不变的变量	260
10.2 样式表中的 XPath	261
10.3 元素	261
10.3.1 元素定义的位置	264
10.3.2 模板及模板的使用	267
10.3.3 判断结构	272
10.3.4 排序的循环	273
10.4 XSLT 函数	274
10.5 XSLT 概念	275
10.6 客户端转换	278
10.7 小结	280
第 11 章 在 Ajax 中应用 XSLT	281
11.1 XSLT	281
11.1.1 神奇的 XML	282
11.1.2 微软公司浏览器的不足	282
11.1.3 XPath 及相关的知识	283
11.1.4 从 Gecko 中所学到的	286
11.1.5 再度提到的问题	290
11.2 表格化的信息	290
11.2.1 只读的	291
11.2.2 可修改的	295
11.3 优点和缺点	295
11.4 小结	296
第 12 章 代码复用	297
12.1 复用=懒惰	298
12.1.1 按代码行计费	298
12.1.2 按页计费	299
12.2 JavaScript 对象	299
12.2.1 集合	301