

<<Ajax设计模式>>

图书基本信息

书名：<<Ajax设计模式>>

13位ISBN编号：9787121039959

10位ISBN编号：7121039958

出版时间：2007-5

出版时间：电子工业出版社

作者：Michael Mahemoff

页数：635

译者：杨仁和

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<Ajax设计模式>>

前言

序言 序 PREFACE 对于想要知道 Ajax 正被如何应用到现实世界的开发者、设计者和经理人，《Ajax设计模式》无疑是为他们量身打造的一份参考资料。

Ajax 是一个新名词，描述行为丰富、类似桌面应用的 Web 应用，这些应用执行在标准的Web浏览器上，而不需要安装任何特别的插件。

它的风行与普及，归功于一些众所瞩目的 Ajax 应用，例如 Gmail、Google Maps和37signals的Basecamp。

在企业应用中，Ajax越来越受欢迎。

本书的模式分成 4 类。

基础技术（Foundational Technology）模式概述了 Ajax 开发所需要的原始技术，例如 XMLHttpRequest 对象。

编程（Programming）模式探索开发者所发现的一些技术，这些技术可以确保他们的 Ajax 应用易于维护且性能良好。

功能性和可用性（Functionality and Usability）模式讨论将在Ajax应用中遇到的，以及 Ajax 使其成为可能的用户界面。

最后，开发（Development）模式阐述了用来监控、调试，以及测试 Ajax 应用的过程。

谁该阅读这本书 Who should Read This Book 如果想要达到以下目的，你应该阅读这本书：
了解 Ajax 能做什么，不能做什么，以及正如何被使用在现实世界里。

加速理解 Ajax 的核心技术，例如XMLHttpRequest、DOM，以及JSON等。

发现开发者用来产生高质量的Ajax架构，改善性能，以及改善可用性的模式。

因为模式本身就是用来参考的，各种背景的读者都可以理解。

想在Ajax领域快速提升的开发者，可以从教程的章节（第2章）及基础技术模式开始。

已经有些 Ajax 开发经验的人，可以从编程模式和开发模式中获益良多。

非技术领域的人，可以以高层的观点，特别是考察功能性和可用性模式以及它们在真实世界中的例子，来看看Ajax能做什么。

为了领会书中的技术性讨论与程序代码范例，程序员应该对网页开发有一定的经验——HTML、表单提交、服务器端脚本，等等。

读者最好也了解一些 JavaScript，因为这本书不是来教你这门编程语言的。

但是本书在写作时，作者确实考虑到了很多读者对 JavaScript 只是基本的了解。

Ajax 主要是关于发生在浏览器里的事，因此本书假设你不知道任何特定的服务器端环境。

在那些涉及服务器端代码的地方，范例基本上都是以PHP作基础，并且用语言中立（language-neutral）的术语来作解释。

谁不该阅读这本书 Who should not Read This Book 如果你没有从事过任何网页开发的工作，在一头栽进Ajax和这些模式之前，或许应该找些基础概念的介绍来看看会更好。

Ajax 开发涉及大范围的技术领域，包括 HTML、XML、CSS、JavaScript 和服务器端开发。

这本书将帮助你理解 Ajax与这些技术的每一项是如何关联的，通常是如何相结合的。

但是，如果你指望彻底学习这些技术中的任何一项，或者你正在寻求有关浏览器细节的参考，建议你查阅更加专业的参考资料。

关于范例 About the Examples 本书的所有范例（教程代码与Ajax模式的代码）都可以从<http://ajaxify.com/run>上下载。

它们已经被开发成与新近版本的Firefox和IE兼容的代码，并且在Firefox 1.5和IE 6.0[注1]上通过了测试。

多数服务器端代码需要PHP 5或更新的版本，并在PHP 5.0.4上也通过了测试。

PHP程序代码应该运行在任何能执行PHP脚本的标准Web服务器上；Apache 1.3.33被用于本书范例的测试。

Wiki的演示需要MySQL，并且在4.1.14-max 的版本上能通过测试。

<<Ajax设计模式>>

关于如何安装演示的信息，请参阅附录B。

浏览器细节 Browser Specifics 实际上对Ajax的目的来说，所有的Ajax模式都能在任何现代的标准浏览器上实现，讨论的焦点应侧重于一般性架构与可用性的问题，而不是浏览器的细节。

但是，可移植性（portability）的问题确实出现在了某些例子中，并且在实现模式具有关键性影响的地方也作了说明（如同本书第二部分，基础技术模式的某些主题中所做的）。

为了不模糊焦点，多数讨论仍局限于IE和Firefox；描绘每个浏览器和版本的风格与手法，超出了模式所要讨论的范围，对此，建议你参考更加专业的资料。

本书的组织 Organization of This Book 第一部分，序曲 最初几章是所有模式的序曲。Ajax简介（第1章）概述了Ajax现象并讨论当前的趋势。

由模式引导的教程（第2章）是Ajax与Ajax模式的教学课程。

Ajax模式背后的设计理念则在Ajax设计：原则与模式（第3章）中讨论，并伴随了模式本身的介绍。

第二部分，基础技术模式 基础技术模式描绘了任何 Ajax 应用的核心“建造模块”（building blocks）。

Ajax本身是一种模式，因此作为第一个模式，Ajax 应用（第4章）扮演了所有 Ajax 模式的基础。

接下来几章探讨创建 Ajax 应用所涉及的 3 项核心技术。

显示操作（第5章）讨论用户界面的重绘，Web Remoting（第6章）整理了几种与服务器通信的做法。

动态行为（第7章）讨论事件——所有衍生自用户或定时机制的事件。

最后，扩展技术（第8章）解释如何以非标准的技术来扩展 Ajax 应用。

第三部分，编程模式 编程模式聚焦于软件的技术性质量，特别是可维护性、稳定性，以及效能。

这部分的第1个章节，即第9章，提供了几种设计 Web Service 的策略。

此外，与 Web Remoting 相关，浏览器-服务器对话（第10章）探讨浏览器与服务器间的信息流，例如，同步技术。

DOM 组装（第11章）涵盖几项策略，处理继服务器响应之后的 注1：确保对所有主要的浏览器（例如 Safari, Opera）完全兼容会很好，特别是对于产品级别的系统而言。

但是在这里，个别的演示并未特别考虑可移植性，每个演示主要是针对特定的 Ajax 概念做介绍。

DOM 组装（DOM population）。

代码生成与重用（第12章）包含了几个实现可维护性与可移植性（portability）的通用编程模式。

最后，性能优化（第13章）不仅讨论了提高更新与通信的速度，也讨论了为无法避免的延迟优化用户的体验。

第四部分，功能性与可用性模式 功能性和可用性模式着重于 Ajax 应用的可用性。

小组件（第14章）介绍一些被整合进很多 Ajax 界面的小组件。

页面架构（第15章）讨论了较高层次的观点，其中的焦点是页面布局、内容分解，以及暴露服务器端的内容。

随着Ajax的普及，因特网正经历一场视觉效果的提升，这对传统的桌面应用系统是前所未有的；最常见的视觉效果将在“视觉效果”（第16章）中描述。

这个部分的最后一章，功能性（第17章）讨论了Ajax使之成为可能的几种新功能。

第五部分，开发模式 开发模式不是你在 Ajax 应用中看到的“具体事物”——不像前面几个部分中的模式——而是能够用来辅助你开发的过程。

诊断分析（第18章）协助进行疑难排查（troubleshooting），以及监测 Ajax 应用程序的运行状态。

测试（第19章），在各个不同的层次上协助进行测试工作，而且是建立在随应用的演进而逐步测试的敏捷实践（agile practice）的基础之上的。

第六部分，附录 共有4个附录。

附录A，最大的附录，是一个Ajax相关框架和库的清单，其中很多对实现本书中的模式大有帮助。

附录B是一组安装代码范例的说明。

附录C介绍模式的一般性理论。

最后，附录D列出了本书所参考的相关资料。

<<Ajax设计模式>>

内容概要

《Ajax设计模式》是一本关于复杂Ajax应用的整体架构设计的图书。好的架构设计应该满足KISS的原则，容易理解和学习、开发效率高、容易维护和扩展。在各种必须要满足的约束条件之下，设计出一个最简化的架构是架构师孜孜不倦追求的目标。幸运的是，经过国外高水平的Web开发专家的探索，这样的架构设计已经有了一套行之有效的模式。这些模式都是围绕着REST的思想来设计的。

RIA+REST将会成为继Web MVC之后，新一代Web应用的首选架构风格。

《Ajax设计模式》是国内出版的第一本深入探讨REST架构设计的书籍。书中的所有模式都有详细的讲解和丰富的代码，理论联系实际，确实是不可多得的一本好书。

<<Ajax设计模式>>

作者简介

作者：（美国）麦赫马夫 译者：杨仁和

<<Ajax设计模式>>

书籍目录

序	第一部分 序曲	第1章 Ajax简介	Ajax与可用性良好的的Web应用	Ajax的崛起	将Web应用Ajax化：Portal的故事	将桌面应用Web化：Office应用的故事	Ajax应用的特性	Ajax技术
		服务器调用之剖析	Ajax的趋势	结论	第2章 由模式引导的教程	Ajax技术概述		
		将Web应用Ajax化：一次一个模式	项目与实践	结论	第3章 Ajax设计：原则与模式	Ajax		
		应用应该具备的属性	为Ajax做设计	Ajax模式概览	剖析模式说明格式	Ajax模式的演示		
		结论	第二部分 基础技术设计模式	第4章 Ajax应用	Ajax应用	第5章 显示操作	显示变形	
		页面重新布置	第6章 Web远程通信	Web服务	XMLHttpRequest调用	IFrame调用		
		HTTP信息流	按需下载的JavaScript	第7章 动态行为	用户动作	计划	第8章 扩展技术	
		富媒体插件	第三部分 编程模式	第9章 Web Services	RESTful服务	RPC服务	Ajax Stub	
		HTML信息	纯文本信息	XML信息	JSON信息	第10章 浏览器-服务器对话	调用	
		跟踪	定期刷新	提交次数限制	明确提交	分布式事件	跨域代理	第11章 DOM
		组装	XML数据岛	浏览器端的XSLT	浏览器端模板	第12章 代码生成与重用	服务器	
		端代码生成	跨浏览器组件	第13章 性能优化	浏览器端缓存	预先获取	推测估计	
		多阶段下载	胖客户端	第四部分 功能性与可用性模式	第14章 小组件	滑动条	进度指示器	
		分级下拉菜单	数据表格	富文本编辑器	输入提示	动态搜索	动态命令行	
		动态表单	第15章 页面架构	拖放	小精灵	弹出框	可改变的内容	微连接
		Portlet	状态区域	更新控制	虚拟工作空间	第16章 视觉效果	一秒钟聚光灯	一
		秒钟变形	一秒钟移动	突出显示	第17章 功能性	延迟注册	直接登录	防主机端保
		存	超时机制	心跳检测	唯一URL	第五部分 开发模式	第18章 诊断	日志
		DOM查看	流量嗅探	第19章 测试	模拟服务	浏览器端测试	服务测试	系统测
		试	第六部分 附录	附录A Ajax框架和库	附录B 设置代码范例	附录C 模式和模式语言	附录D 参	
		考文献	索引					

<<Ajax设计模式>>

媒体关注与评论

序言序PREFACE对于想要知道 Ajax 正被如何应用到现实世界的开发者、设计者和经理人,《Ajax设计模式》无疑是为他们量身打造的一份参考资料。

Ajax 是一个新名词,描述行为丰富、类似桌面应用的 Web 应用,这些应用执行在标准的Web浏览器上,而不需要安装任何特别的插件。

它的风行与普及,归功于一些众所瞩目的 Ajax 应用,例如 Gmail、Google Maps和37signals的Basecamp。

在企业应用中,Ajax越来越受欢迎。

本书的模式分成 4 类。

基础技术 (Foundational Technology) 模式概述了 Ajax 开发所需要的原始技术,例如 XMLHttpRequest 对象。

编程 (Programming) 模式探索开发者所发现的一些技术,这些技术可以确保他们的 Ajax 应用易于维护且性能良好。

功能性和可用性 (Functionality and Usability) 模式讨论将在Ajax应用中遇到的,以及 Ajax 使其成为可能的用户界面。

最后,开发 (Development) 模式阐述了用来监控、调试,以及测试 Ajax 应用的过程。

谁该阅读这本书 Who should Read This Book 如果想要达到以下目的,你应该阅读这本书:了解 Ajax 能做什么,不能做什么,以及正如何被使用在现实世界里。

加速理解 Ajax 的核心技术,例如 XMLHttpRequest、DOM,以及JSON等。

发现开发者用来产生高质量的Ajax架构,改善性能,以及改善可用性的模式。

因为模式本身就是用来参考的,各种背景的读者都可以理解。

想在Ajax领域快速提升的开发者,可以从教程的章节 (第2章) 及基础技术模式开始。

已经有些 Ajax 开发经验的人,可以从编程模式和开发模式中获益良多。

非技术领域的人,可以以高层的观点,特别是考察功能性和可用性模式以及它们在真实世界中的例子,来看看Ajax能做什么。

为了领会书中的技术性讨论与程序代码范例,程序员应该对网页开发有一定的经验——HTML、表单提交、服务器端脚本,等等。

读者最好也了解一些 JavaScript,因为这本书不是来教你这门编程语言的。

但是本书在写作时,作者确实考虑到了很多读者对 JavaScript 只是基本的了解。

Ajax 主要是关于发生在浏览器里的事,因此本书假设你不知道任何特定的服务器端环境。

在那些涉及服务器端代码的地方,范例基本上都是以PHP作基础,并且用语言中立 (language-neutral) 的术语来作解释。

谁不该阅读这本书 Who should not Read This Book 如果你没有从事过任何网页开发的工作,在一头栽进Ajax和这些模式之前,或许应该找些基础概念的介绍来看看会更好。

Ajax 开发涉及大范围的技术领域,包括 HTML、XML、CSS、JavaScript 和服务器端开发。

这本书将帮助你理解 Ajax与这些技术的每一项是如何关联的,通常是如何相结合的。

但是,如果你指望彻底学习这些技术中的任何一项,或者你正在寻求有关浏览器细节的参考,建议你查阅更加专业的参考资料。

关于范例 About the Examples 本书的所有范例 (教程代码与Ajax模式的代码) 都可以从 <http://ajaxify.com/run> 上下载。

它们已经被开发成与新近版本的Firefox和IE兼容的代码,并且在Firefox 1.5和IE 6.0[注1]上通过了测试。

多数服务器端代码需要PHP 5或更新的版本,并在PHP 5.0.4上也通过了测试。

PHP程序代码应该运行在任何能执行PHP脚本的标准Web服务器上;Apache 1.3.33被用于本书范例的测试。

Wiki的演示需要MySQL,并且在4.1.14-max 的版本上能通过测试。

<<Ajax设计模式>>

关于如何安装演示的信息，请参阅附录B。

浏览器细节Browser Specifics实际上对Ajax的目的来说，所有的Ajax模式都能在任何现代的标准浏览器上实现，讨论的焦点应侧重于一般性架构与可用性的问题，而不是浏览器的细节。

但是，可移植性（portability）的问题确实出现在了某些例子中，并且在实现模式具有关键性影响的地方也作了说明（如同本书第二部分，基础技术模式的某些主题中所做的）。

为了不模糊焦点，多数讨论仍局限于IE和Firefox；描绘每个浏览器和版本的风格与手法，超出了模式所要讨论的范围，对此，建议你参考更加专业的资料。

本书的组织Organization of This Book第一部分，序曲最初几章是所有模式的序曲。

Ajax简介（第1章）概述了Ajax现象并讨论当前的趋势。

由模式引导的教程（第2章）是Ajax与Ajax模式的教学课程。

Ajax模式背后的设计理念则在Ajax设计：原则与模式（第3章）中讨论，并伴随了模式本身的介绍。

第二部分，基础技术模式基础技术模式描绘了任何 Ajax 应用的核心“建造模块”（building blocks）。

Ajax本身是一种模式，因此作为第一个模式，Ajax 应用（第4章）扮演了所有 Ajax 模式的基础。

接下来几章探讨创建 Ajax 应用所涉及的 3 项核心技术。

显示操作（第5章）讨论用户界面的重绘，Web Remoting（第6章）整理了几种与服务器通信的做法。

动态行为（第7章）讨论事件——所有衍生自用户或定时机制的事件。

最后，扩展技术（第8章）解释如何以非标准的技术来扩展 Ajax 应用。

第三部分，编程模式编程模式聚焦于软件的技术性质量，特别是可维护性、稳定性，以及效能。

这部分的第1个章节，即第9章，提供了几种设计 Web Service 的策略。

此外，与 Web Remoting 相关，浏览器-服务器对话（第10章）探讨浏览器与服务器间的信息流，例如，同步技术。

DOM 组装（第11章）涵盖几项策略，处理继服务器响应之后的注1：确保对所有主要的浏览器（例如 Safari，Opera）完全兼容会很好，特别是对于产品级别的系统而言。

但是在这里，个别的演示并未特别考虑可移植性，每个演示主要是针对特定的 Ajax 概念做介绍。

DOM 组装（DOM population）。

代码生成与重用（第12章）包含了几个实现可维护性与可移植性（portability）的通用编程模式。

最后，性能优化（第13章）不仅讨论了提高更新与通信的速度，也讨论了为无法避免的延迟优化用户的体验。

第四部分，功能性与可用性模式功能性和可用性模式着重于 Ajax 应用的可用性。

小组件（第14章）介绍一些被整合进很多 Ajax 界面的小组件。

页面架构（第15章）讨论了较高层次的观点，其中的焦点是页面布局、内容分解，以及暴露服务器端的内容。

随着Ajax的普及，因特网正经历一场视觉效果的提升，这对传统的桌面应用系统是前所未有的；最常见的视觉效果将在“视觉效果”（第16章）中描述。

这个部分的最后一章，功能性（第17章）讨论了Ajax使之成为可能的几种新功能。

第五部分，开发模式开发模式不是你在 Ajax 应用中看到的“具体事物”——不像前面几个部分中的模式——而是能够用来辅助你开发的过程。

诊断分析（第18章）协助进行疑难排查（troubleshooting），以及监测 Ajax 应用程序的运行状态。

测试（第19章），在各个不同的层次上协助进行测试工作，而且是建立在随应用的演进而逐步测试的敏捷实践（agile practice）的基础之上的。

第六部分，附录共有4个附录。

附录A，最大的附录，是一个Ajax相关框架和库的清单，其中很多对实现本书中的模式大有帮助。

附录B是一组安装代码范例的说明。

附录C介绍模式的一般性理论。

最后，附录D列出了本书所参考的相关资料。

本书的格式约定Conventions Used in This Book斜体 用来表示模式名称。

等宽字体用来表示程序范例或片段。

<<Ajax设计模式>>

等宽粗体用来表示重要的代码。

“浏览器”这个词，是浏览器端整个系统的简化表示——不只是浏览器应用（例如 Firefox），还包括运行在上面的Web应用（例如 Google Maps）。

当特指浏览器应用时，通常会把它称为“Web浏览器”——无论如何，从上下文来看，这是相当明显的。

同样地，“服务器”或“服务器端”指的是服务器上的所有东西——操作系统（例如Linux），Web服务器（例如Apache），以及运行在上面的特定于应用的脚本。

“ECMAScript”和“JScript”这两个词并未出现在本书中，“JavaScript”被用来涵盖这些名称。

“类”这个词，被用来描述以类似面向对象系统中类的方式使用的JavaScript函数。

出现在每个模式里的第一个图形，是该模式的图解。

本书所使用的公共首字母缩略词如下所示：CSS：层叠式样式表（Cascading Style Sheets）。

DHTML：动态HTML（Dynamic HTML）。

DOM：文档对象模型（Document Object Model）。

HTML：超文本标记语言（Hypertext Markup Language）。

IE：微软 Internet Explorer。

XML：可扩展标记语言（eXtensible Markup Language）。

代码范例的格式约定Conventions in Code Examples设计图解基于统一建模语言（UML）表示法，但在多数情况下，这并不是很正式。

在代码范例中，.phtml后缀用于输出HTML的PHP文件；所有其他的PHP文件（例如商业逻辑模块）则以.php结尾。

如第2章所述，`$()`是`document.getElementById()`的别名，用来减少代码的凌乱程度（受prototype库启发——见<http://prototype.conio.net/>）。

为清晰起见，某些代码范例重新进行了格式化。

联系我们How to Contact Us我们已尽力核验本书所提供的信息，尽管如此，仍不能保证本书完全没有瑕疵，而网络世界的变化之快，也使得本书永不过时的保证成为不可能。

如果读者发现本书内容上的错误，不管是赘字、错字、语意不清，甚至是技术错误，我们都竭诚虚心接受读者指教。

如果您有任何问题，请按照以下的联系方式与我们联系。

奥莱理软件（北京）有限公司北京市 海淀区 知春路49号 希格玛公寓B座809室 邮政编码：100080 网页：<http://www.oreilly.com.cn> E-mail：info@mail.oreilly.com.cn 与本书有关的在线信息（包括勘误、范例程序、相关链接）如下所示。

<http://www.oreilly.com/catalog/ajaxdp/>（原书）<http://www.oreilly.com.cn/book.php?bn=978-7-121-03995-9>（中文版）博文视点资讯有限公司（武汉分部）湖北省 武汉市 洪山区 邮科院路特1号 湖北信息科技大厦1406室 邮政编码：430074 电话：（027）87690813 传真：（027）87690813转817 读者服务网页

：<http://bv.csdn.net> E-mail：shenguang@broadview.com.cn（读者信箱）broadvieweditor@gmail.com（投稿信箱）致谢Acknowledgments在线撰写这本书意味着：在整个过程中，我接收到了很多很棒的反馈，因此有很多应该感谢的人。

但换句话说，我很有可能遗漏了某些人！

如果情况真是如此，请来信告知，我会加以改正。

我的编辑，Simon St.Laurent，总是对我的疑问迅速做出响应，并且以一种不寻常的方式将一本写作中的书处理得很棒。

也感谢本书的校稿人Mary Brady巨细靡遗的工作，以及插画家Rob Romano，感谢他的协助，让本书的内容一目了然。

也要感谢欧莱礼接受Ajax Patterns这个网站，特别是让我在线写作这本书，还有内容毫无限制的博客与播客。

这本书从Ajax Patterns（<http://softwareas.com/ajax-patterns>）上的博客中成长，Thomas Baekdal拥有Ajax可用性原则的洞察力，启发了最初发表的几个模式（<http://www.baekdal>）。

<<Ajax设计模式>>

关com/articles/Usability/XMLHttpRequest-guidelines)。

也要谢谢Brent Ashley, 感谢他在最初的Ajax会议以及他的博客里关于模式的讨论, 很明显地, 这引发了对于这个项目的在线工作的最初关注。

这本书的想法也要大大归功于Jesse-James Garrett和他的Ajax文章, 没有这些文章, 就没有这本书。

本书所有的审阅者, 在整个写作过程中, 对在线版本提供了很多有价值的反馈, 对于本书的草稿版本, 更提供了许多巨细靡遗的意见。

这些审阅者是: Cameron Shorter和Kevin Vinsen (ADI Limited, Mapbuilder), 还有他们的雇主, ADI公司, 给予他们两人审阅本书的时间; Jep Castelein (Backbase)、Daniel Czarnecki (Zoltak Consulting)、Tony Hill (Thomson Corporation), 以及 Alex Kirk (Blummy.com, WizLite.com)。

另外, 深深感谢位于Urbana-Champaign的Illinois大学的Software Architecture Group (SAG) 的成员, 感谢他们主导一些关于模式的内容丰富的讨论, 并将讨论内容的声音文件公开在网络上

(<http://www.softwareas.com/sag-ajax-patterns-review-1>) ——这些宝贵的意见最后都融入到了本书的内容里。

领导这个团体的Ralph Johnson, 推荐Ajax Patterns作为讨论的对象, 还有Brian Foote, 在整个过程中他让我获得最新的信息。

我也感激加入 AjaxPatterns.org Wiki 公共部分的每个人, 他们通过电子邮件、博客, 以及 Wiki 上的讨论, 提供对本书在线草稿的反馈。

因为模式是从很多现有应用中发现的, 我想感谢所有这些范例的创造者, 他们是真正的先锋。

由衷感谢对我解释他们的设计思想的那些人: Kevin Arthur (Stream)、Richard Cowin 与 Bill Scott (OpenRico)、Donovan Preston (LivePage)、Jeremy Ruston (TiddlyWiki), 以及 Tino “Crisp” Zijdel (DHTML Lemmings)。

这些模式也是基于很多文章、范例代码、想法、概念证明, 以及直接的建议的。

很多模式之所以会出现在书中, 特别是因为下列作者和开发者 (也会在个别的模式描述中提及) 所有的贡献: Julien Couvreur、James Dam、Abe Fettig (Twisted、Jotspot)、Chris Justus、Christopher Kruslicky、Alex Russell (Dojo、Jotspot)、Richard Schwartz、Mike Stenhouse、Joel Webber, 以及Chris Were。

另外, 有几个以新闻为焦点的服务, 使得当相关内容出现时, 寻找到这些内容变得容易得多 (也帮助别人找到了AjaxPatterns.org!

)。这些资源有: Niner Niner 的 AjaxBlog.com、Chris Cornutt的AjaxDeveloper.org (Chris目前在Ajaxian.com)、Ajaxian.com的Dion Almaer、Ben Galbraith与Rob Sanheim、Shane Witbeck的 AjaxMatters.com、Mike Papageorge的FiftyFourEleven.com、Brian Benzinger 的 SolutionWatch.com, 以及 Mike Arrington 的 Techcrunch.com。

最后, 但并不是最不重要的, 满怀的感激要献给我的家人, 感谢他们在我的整个写作过程中的支持。审校者序传统的DHTML技术与XMLHttpRequest、IFrame等Web Remoting技术相结合之后, 得到了服务器端强有力的支持, 这使得DHTML技术如同插上了腾飞的翅膀, 并孕育出了一种被称作Ajax的新技术。

自从Ajax技术出现之后, 对于其本质和如何以最有效的方式来使用它的研究一直在持续进行着, 其目的是使Web应用获得在可用性、用户体验和开发效率等方面的最大利益。

这一切的研究都是建立在现有实践的基础之上的。

实践总是远远走在理论的前面, 这是广大富有创造性、勇于开拓的Web开发者的功劳。

互联网经济的复苏造就了Web 2.0的神话, Web 2.0则促进了Ajax技术的发展。

在Web 2.0网站中, 对于Web应用的交互设计提出了各种苛刻的要求, 这促使Web开发者不断去探索新的未知领域, 探索基于Web标准的技术所能达到的极限。

Web开发中新的需求层出不穷, 这为他们发挥自己的创意、实现自己的梦想创造了广阔的舞台。

今天, 浏览器中的JavaScript通过SVG、Canvas等技术获得了全面的绘图能力, 可以用来绘制图表和开发视频游戏。

<<Ajax设计模式>>

浏览器的最新发展还为JavaScript脚本提供了更大的本地存储空间，远远超出了Cookie的4KB数据量的限制。

这非常有利于开发复杂的Ajax应用（例如online的字处理器），这些应用需要在客户端保存大量的状态。

Ajax应用可以得到Flash的协助，可以通过Flash的能力来播放多媒体内容。

Ajax应用还可以通过HTTP信息流（另外一个名称叫做Comet）的方式与服务器通信，在获得了更好的实时性的同时，也消耗了更少的网络流量和服务器端资源。

通过这种方式可以实现信息的实时推送，这种技术可以被应用在股票信息实时发布和即时通信等广泛的领域中。

今天的Web开发与3年前相比已经有了非常大的不同，新的事物仍在不断涌现。

如果有一个见多识广、富有洞察力的人将这些技术系统地整理起来，对广大Web开发者开阔眼界，提高技能会有非常大的帮助。

《Ajax设计模式》这本书恰逢其时，刚好填补了这样一个巨大的空缺。

在书中，作者Michael Mahemoff以最广阔的视野，最大努力，将当今已经出现在Web上的各种基于Ajax的开发技术进行了一个全景式的展示，系统地将这些技术以模式的形式加以整理和编目，并辅以精心设计的代码。

本书是对于Ajax技术成型近两年以来所取得成果的一个全面总结。

所有的Web开发者都可以从本书中受益，学习书中所列出的众多的开发技术和开发套路，并尽快将这些优秀的技术应用在自己的项目中。

古龙的《多情剑客无情剑》（又名《小李飞刀》）是大家熟悉的武侠小说。

其中有一人叫做百晓生，这个人似乎无所不知无所不晓。

他编写了一个兵器谱，对于江湖中最厉害的兵器进行了排序。

小李飞刀排名第三，因此李寻欢又被称做小李探花。

《Ajax设计模式》的作者就是Ajax世界中的百晓生。

他是著名的Ajax站点AjaxPatterns网站的站长，本书的内容是以在AjaxPatterns网站建立一年半以来所开展的讨论为基础的。

自从GoF的《设计模式》一书出版后，将各个技术领域中之有效的一些通用的解决方案总结为模式，成了一种对于该领域进行研究的最有效的方法。

一个技术领域出现了大量的模式，说明对于这个领域的研究已经进入了高级阶段，这个领域的技术已经发展成熟。

现在，对于Ajax领域的研究也进入了这样的一个高级阶段。

不过Ajax技术距离完全成熟还很远，在近几年之内仍然会取得迅速的发展。

Martin Fowler在面向对象分析和设计（OOAD）领域做出了开创性的杰出贡献，Michael Mahemoff对于Ajax领域也做出了类似的贡献。

也许无法相提并论，但是他使得对于Ajax技术的研究纳入了理论化、系统化的快车道，非常有助于将来对于Ajax技术进行更加深入的研究。

基于Web标准的开发技术仍然会统治互联网，主要原因有两个：1. 基于这些技术可以提供目前Web应用可获得的、最好的可用性。

国外对于Web可用性的研究已经有很多年，得到了很多行之有效的设计原则和最佳实践。

W3C对于提高Web应用的可用性也非常重视，在上世纪90年代就建立了Web Accessibility Initiative（WAI，<http://www.w3.org/WAI/GL/>）。

美国在Web应用的可用性方面还有相关的立法（<http://www.section508.gov/>）。

ISO也正在制订Web可用性方面的标准（<http://www.userfocus.co.uk/articles/ISO23973.html>）。

所以，Web应用要达到很高的可用性，目前唯一的选择就是基于Web标准的技术。

如果把这些研究成果和最佳实践统统扔掉，改用其他的RIA技术，都会影响Web应用的可用性。

可用性与界面的华丽是两个不同的问题。

2. 这些技术对于搜索引擎是最友好的，便于进行搜索引擎优化。

<<Ajax设计模式>>

假设我在网络上从事电子商务。

如果采用Web标准之外的技术，很可能会严重影响我的网站对于搜索引擎的可视性（visibility），这会严重影响我的业务。

因为目前绝大多数通用的搜索引擎都是围绕基于Web标准的技术而设计的，它们搜索HTML内容最为有效，如果是Flash内容或者其他内容，很有可能无法被搜索到。

所以我判断基于Web标准的技术仍然会统治互联网（Internet）。

而其他的RIA技术呢，主要将占据企业内部应用（Intranet）的市场。

Ajax技术将来的发展趋势我预测会有这些：1. 各种Ajax组件库和框架逐渐成熟，提供了更多功能丰富的组件。

使用这些组件做开发可以改善Web应用的可用性，并且可以大幅提高开发效率。

2. 各种Ajax组件库和框架会逐渐融合，并且改善它们之间的互操作性。

3. JavaScript 2.0推出并且逐渐被广泛采用。

其所提供的新功能对于基于JavaScript的大规模组件化开发非常有价值。

4. Ajax开发工具逐渐成熟，出现功能全面的IDE，支持JavaScript的代码补全、重构、调试、单元测试、部署等功能。

5. 浏览器本身也有可能实现一些将来被广泛应用的Ajax组件，例如Tree、DataGrid等。

作为《Ajax设计模式》一书的审校者，我有幸先于国内的读者读到本书的内容。

我的感觉就像是在森林深处发现了一个巨大的宝藏，阿里巴巴初次见到宝藏的那种感觉。

我原先所理解的Ajax技术所能达到的范围一下子被拓宽了很多倍。

书中大量有趣的技术，就像是一件件闪着璀璨亮光的珍宝，我都想尽快去尝试一番，感觉我似乎一下子变成了一个富翁。

《Ajax设计模式》一书是Ajax开发者的百宝箱，我们怀着虔诚的心情，将本书奉献给国内的Web开发者。

希望读者在阅读本书的过程中能够获得与我们相同的体验，享受到巨大的快乐。

Web开发并不是一场苦功，而是一件很快乐的事情，让我们与国外的Web开发专家一道，尽力使它变得更加有趣。

本书由ajaxcn.org负责审校。

李锜负责审校第1~5、10、11章和正式章节之外的所有文字，包一磊负责第8~10章，俞黎敏负责第13~17章，张祖良负责第6、7、18、19章。

由于能力有限，难免会出现一些错误，敬请读者指正，我们一定会在后续的印刷中尽快改正。

李 锜2007年元宵节 于上海

<<Ajax设计模式>>

编辑推荐

Ajax，即异步JavaScript与XML，目前是网络开发人员中最热门的编程技术之一。Ajax综合了大量技术，通过在浏览器中使用JavaScript来处理网络服务的响应，为创建基于XML或者JSON网络服务的交互式网络应用程序提供了坚实的基础。

《Ajax设计模式》为你展示了一些最佳实践方法，这些方法能够迅速提升你的网络应用程序开发项目。

这《Ajax设计模式》分析了其他人如何成功地处理各种相互矛盾的设计原则，并将有用的信息直接传递给你。

《Ajax设计模式》中描述的模式分为以下4种类别：基本技术：检阅Ajax开发所必需的原始技术；编程：揭示开发者所发现的、可以保证他们的Ajax应用易于维护的技术；功能和可用性：描述在Ajax应用中将会是到的用户界面的类型，以及Ajax技术使其成为可新的新功能；开发：解释用于监控、调试和测试Ajax应用的开发过程。

《Ajax设计模式》将使你对核心的Ajax技术了如指掌，例如XMLHttpRequest、DOM、JSON等。紧随在技术讨论之后的是实例，这样你就能看到对于Ajax，什么是可能的，什么是不可能的。

这个便捷的参考将会帮助你制造出高品质的Ajax架构，使得Web应用性能流畅，并且改善用户的体验。

Michael Mahemoff在墨尔本大学获得了计算机科学和软件工程的哲学博士（Ph.D）学位，他的论文是“软件工程中的设计重用和人机交互”。

他居住在伦敦，为银行、卫生保健、后勤等行业做软件开发问题的技术咨询。

AJAX三剑客之模式之剑，全景展现Ajax技术结晶，全面总结Web2.0开发利剑。

<<Ajax设计模式>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>