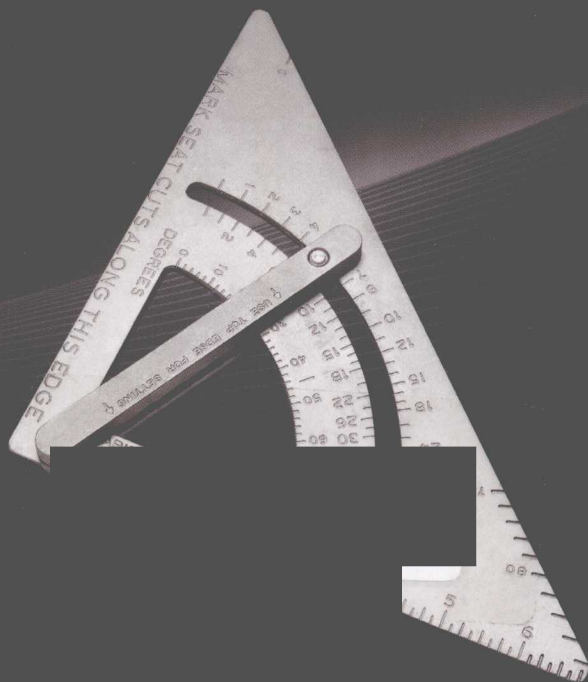


面向服务的AJAX 架构模式与应用开发

[美] Daniel Larson 著
李庆松 徐小珑 邹佳 译



人民邮电出版社
POSTS & TELECOM PRESS

面向服务的AJAX 架构模式与应用开发

[美] Daniel Larson 著
李庆松 徐小珑 邹佳 译

人民邮电出版社

北京

图书在版编目 (C I P) 数据

面向服务的AJAX架构模式与应用开发 / (美) 拉尔森
(Larson, D.) 著 ; 李庆松, 徐小珑, 邹佳译. — 北京 :
人民邮电出版社, 2011. 1
ISBN 978-7-115-23889-4

I. ①面… II. ①拉… ②李… ③徐… ④邹… III.
①计算机网络—程序设计 IV. ①TP393.09

中国版本图书馆CIP数据核字 (2010) 第197247号

内 容 提 要

今天, AJAX 已经成为富互联网应用的核心技术, 改变用户体验的同时, 也改变了程序员的生活, 给他们以更大的挑战。

微软公司推出.NET 平台已有 10 年, 这项技术已经取得了巨大的成就, 同样也面临 AJAX 的挑战。

本书面向使用微软技术的中高端的 Web 编程人员, 基于 VS 2008 中的 AJAX 框架库进行详细的讲述, 同时配以大量详尽的代码示例, 阐述了一种以 AJAX 为浏览器前端技术, 以 WCF 为后端 Web 服务平台的编程架构——即面向服务的 AJAX 架构。

本书采用了从架构到细节, 从技术到应用的讲述方式, 先介绍了面向服务的体系架构的 AJAX 与 WCF 原理, 然后从实践的角度讲述了微软 AJAX 库的技术细节, 最后讲述了面向对象的 JavaScript 控件库和基于微软 AJAX 库的应用。

本书论述仔细, 脉络清晰。适合于使用微软技术编程的开发与学习人员, 可以作为开发人员的案头参考, 也可用作培训教材。

面向服务的 AJAX 架构模式与应用开发

- ◆ 著 [美] Daniel Larson
- 译 李庆松 徐小珑 邹 佳
- 责任编辑 刘 浩
- ◆ 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>
北京艺辉印刷有限公司印刷
- ◆ 开本: 800×1000 1/16
印张: 17.75
字数: 562 千字 2011 年 1 月第 1 版
印数: 1-2 500 册 2011 年 1 月北京第 1 次印刷

著作权合同登记号 图字: 01-2009-5544 号

ISBN 978-7-115-23889-4

定价: 49.00 元

读者服务热线: (010) 67132705 印装质量热线: (010) 67129223
反盗版热线: (010) 67171154

序

本书将给你带来大量有关利用微软技术来构建 AJAX 应用的技术信息。在深入细节之前，请抽出一点时间和我们一起探讨一下为什么你想这样做以及思考它的方法。

AJAX，这个名字刚流行没多久，但是这项技术，已经应用相当长的时间了。早期版本的微软 Outlook 的网络访问在一定程度上开创了先河。尽管仅限于 IE 浏览器并使用了私有的扩展，但是这已经说明 Web 应用可以做更多的事情，并且并不是所有的功能实现都需要页面视图。从那之后，很多程序也采用了这项技术。

AJAX 已然成为时下的流行语。如果你的应用没有使用它，开发者们就会说“如果采用 AJAX 技术会好很多”；市场人员们会说“我们需要添加一些 AJAX 功能”；技术媒体会说“非常不 AJAXy”；狂热的技术爱好者则会说“这根本就是 Web 1.0 嘛！”。其实客户不会使用这个词，他们很可能就根本没听说过这个词，除非他们是技术爱好者。西尔维娅姑妈就不会担心这些东西，你问她什么是“页面视图”，她会很诧异的答案你“这些跟我有什么关系？”。

用户关注的是体验，这也就是为什么一些成功的应用采用了 AJAX。提升用户体验以及提高运行效率（对用户或者你的应用）无疑是最高目标，显然 AJAX 有助于实现这些目标。

本书的大部分将会讨论面向服务的架构和如何设计好的服务以便与你的 Web 应用更好更协调地工作。即使不使用 AJAX 的情况下，按照此种方式设计应用也会给你带来不少便利，并且通过介绍 WCF 类似的技术，应用会变得更简单。设计服务使得你不得不认真考虑如何将你的应用划分为不同的模块，以什么样的方式把这些模块关联起来并能减少应用与浏览器之间的通信次数。

许多 AJAX 应用会将一些处理过程从服务器转嫁给客户端，这时必须谨慎对待。在做设计时，需要考虑如下几个因素。

- 服务器 CPU（数据库，应用服务器，Web 服务器）
- 网络利用率
- 客户端 CPU（浏览器占有的）

除上述因素外，还有一些其他因素，例如本地浏览器存储等，但是上述 3 个因素是最为关键的。只要打开应用中的链接，这几个因素就会相互影响。

服务器端 CPU 不难掌握——用户了解自己的数据库和应用逻辑，对哪些资源昂贵哪些廉价会有一个大致的认识。如果顺利的话，通过灵活地设计服务，可以将问题化简为更小更容易处理的问题，使得整个服务器的管理更加可控。

网络利用率是要慎重考虑的。在高速数据连接的当今世界，那些使用蜂窝网络的用户很容易被忽略掉。比如一些 3G 网络，具有非常快的吞吐量却可能会出现糟糕的延迟。如果想确定这些网络是以何种方式影响你的应用，唯一的办法就是测试！

客户端 CPU 有点麻烦。在排名前四位的桌面浏览器上，JavaScript 的执行效率相差有十倍之多，也就是说常用的最快的浏览器执行 JavaScript 比最慢的浏览器快十倍（根据公布的

测试结果),更不用提移动浏览器了,那是绝对的慢。在所有你能找到的浏览器上测试!曾经我就运行过这样一个应用,在一种浏览器上运行得相当平稳,可是在另外一种浏览器中运行时,总会使客户端机器死机。真是很难预测的——只有测试!

在 NewsGator,我们的企业级产品线都使用到了 AJAX,用到了本书中所述的很多技术,包括面向服务的架构。这样的方案能使我们实现我们想要的用户体验,限制对服务器耗费资源的回调 API 的数量,实现对终端用户的异步更新,甚至通过消除长时间页面加载提升一些内置在 SharePoint 中的功能的可用性。

不管你读不读这本书,在构建 Web 应用时,最好将它们发挥得淋漓尽致——使用、敏捷、直观和有吸引力。使用 AJAX 来解决问题,使用 AJAX 为用户把产品做得更好。它就是你工具箱中的工具,而且是一个强大的工具。

Greg Reinacker

NewsGator 技术公司 创始人兼 CTO

前言

AJAX 从根本上改变了我们创建和使用 Web 应用程序的方式。与我在 1992 年首次创建使用 Web 页面时相比,重新加载页面,提交表单甚至网站导航已经发生了翻天覆地的变化。今天,客户期望的是具有实时数据的互联网应用,而不是只有静态数据的简单页面。随着越来越多的商业应用转移到互联网上,以及像谷歌 Chrome 这样能进一步优化互联网应用浏览器的出现,这一期望变得更加真实。

前些年 AJAX 还是一个崭新的概念。大多数人还只是看到 AJAX 以吸引眼球的方式应用在像谷歌地图和微软 Windows Live Local 这样的应用中。只有极少数的开发者知道如何将它应用在自己的应用程序中,当时已经算是奇迹。但是时至今日,没有 AJAX 的应用会让人觉得迟钝、笨重、过时。

另外需要指出的是 AJAX 不是用户体验的代名词。创建一个非面向服务的应用,你仍然能体会到一些 AJAX 所带来的用户体验的优势,但不会体验到纯正 AJAX 解决方案所带来的灵活性和持久性。在本书中不会涉及 Update Panel——ASP.NET AJAX 中的一个 Web 组件,用来通过 ASP.NET 回传模式实现“神奇”的类 AJAX 行为。

AJAX 所展现的是一个架构模式,而不是纯粹的用户体验:异步 JavaScript 和 XML。另一种方法来描述这种模式是这样的:“编写一些 Web 服务,然后开发相应的 JavaScript 库来使用它们。”这本书采用的也是这种方式,也可称之为面向服务的 AJAX。

面向服务的 AJAX 采用以 Web 服务 API 为入口的方式。这些 API 实现了应用的核心功能并将应用暴露给多个客户端。在这种方式下,很容易实现在不同的客户端之间交换数据,不论 Flash 和 Silverlight,还是远程的 Windows 或者 Macintosh 客户端。因为我们采用的是面向服务的方式,相同的服务可以被扩展为可供 AJAX 应用使用的远程 API 和 JavaScript API。AJAX 中的 X 表示 XML,数据格式并不重要,WCF 既支持 XML 也支持 JavaScript 序列化(JSON)数据流。

本书中,我采用的是微软的 AJAX 库(ASP.NET AJAX 的客户端库)实现客户端运行程序。微软 AJAX 库贯穿于全书之中,并在第 4 部分的第 8 章做详细的介绍。微软 AJAX 库用来创建客户端运行程序,通过 JavaScript 库实现与后台程序的 Web 服务异步调用。

多年前我开始第一次实现这种模式的时候,使用的是 ASMX Web 服务和 HTTP 处理程序的组合实现 API,利用微软 ASP.NET AJAX 扩展集成 JavaScript 来支持这些服务。后来出现了 WCF,再后来微软 .NET Framework 3.5 引入了 WCF Web 编程模型。一切都随之改变,通过进一步的抽象,服务开发者不需要指定这些服务如何提供接口——相同的服务可以通过一大堆的绑定和行为进行扩展,而且从任何地方都可以调用它们:JavaScript 客户端、TCP/IP 绑定甚至是消息队列集成。

正如一个设计糟糕的 ASP.NET 应用程序很难维护一样,一个粗陋不堪的 AJAX 应用将会成为开发者的梦魇。转向到 AJAX 会有很多挑战:浏览器兼容性问题,可访问性问题以及将

客户接口逻辑从服务器移植到客户端的架构设计挑战。实质上我们要做的是将客户接口从 ASP.NET 中的 C#、VB 这些编译型语言移植到像 JavaScript 这种运行在客户端的解释型语言上。但是一个设计完善的基于服务的架构，将会使后端应用比想象的更加灵活和安全，你是在为未来而创建。应用程序不会被 ASP.NET、AJAX、Silverlight 这样的前端技术所束缚——以使用服务的形式被创建。这将大大降低前端程序开发的负担——相比于传统的 Web 应用，你可以以较少的花费添加功能或集成多个组件。

掌握了基于服务的 AJAX 架构以及微软富互联网应用框架，包括 ASP.NET AJAX 和 WCF，开发人员可以创建企业级的比传统 Web 应用更易编写、更易维护并且面向未来的应用。主程序不会与用户接口绑定，但是会以编译代码的形式保留在服务器上。在后端，可以使用 WCF 以服务的方式提供应用数据，服务可被多个应用使用，而不只是你的 AJAX 应用。WCF 是另外一项向未来迈了一大步的技术——用来解耦应用和接口。技术在演进，你不会继续坚持落后的服务平台吧。

AJAX 开发使用你已有的技术和 ASP.NET 的知识，但是也确实需要一些 Web 开发的新视野。本书的目的就是使用现有的开发技术以面向服务的 AJAX 架构模式来开发新型的应用。

本书面向的读者

本书面向的是使用微软技术的中高级 Web 开发人员。本书指导开发人员从 ASP.NET 迁移到使用 WCF Web 服务的 ASP.NET AJAX 上来。本书包含部分 C# 代码，其他主要为 JavaScript 代码。VB.NET 开发人员也很容易转换 C# 代码，因为本书关注的是架构。本书代码示例使用 C# 编写，通过 VS 2008 在 .NET Framework 3.5 SP1 下编译通过，不过你也会发现很多 JavaScript 代码。读者需要具备一些 C#、ASP.NET、Web 开发以及 .NET Framework 的基础知识。WCF 的知识虽不是必须的，但掌握它会有非常大的帮助。我会为 AJAX 开发者介绍一些 WCF 的基础知识，并在介绍时给出一些有意义的参考。

本书的内容

本书关注的是利用微软 .NET Framework 3.5 和微软企业服务器使用面向服务的 AJAX 模式开发软件。

第 1 部分“AJAX 与 WCF”。这部分介绍的是面向服务的 AJAX 架构模式和作为 Web 服务技术平台的 WCF。

- 第 1 章“面向服务的 AJAX 基本原理”，介绍面向 AJAX 应用的面向服务的架构模式。对模式进行概述并给出简单例子便于读者开始学习。

- 第 2 章“AJAX 应用服务器端：面向服务与 WCF”，介绍作为 Web 服务技术平台的 WCF 以及关注于现代软件架构中面向服务原则。

- 第 3 章“AJAX 应用服务器端：WCF 的 Web 编程模型”，阐述 WCF 的 Web 编程模型和 REST 架构风格以及 WCF 框架中的联合服务。

第2部分“AJAX 实践”。本部分关注的是客户端编程，使用微软 AJAX 库的面向服务的 AJAX 应用。

- 第4章“采用微软 AJAX 库的 AJAX 运行程序”，介绍微软 AJAX 库并关注核心运行控件和部署。
- 第5章“微软 AJAX 库”，深入探视微软 AJAX 库，重点关注核心运行组件，包括主要命名空间，基本类型和 JavaScript 类型系统。
- 第6章“使用 Sys.Services 实现 AJAX 应用服务”，涉及支持 AJAX 鉴权，配置以及利用 ASP.NET AJAX 框架实现鉴权的应用服务。

第3部分“应用 AJAX”。本部分关注的是在使用面向服务的 Web 服务框架时，实现面向对象的 JavaScript 控件库和基于微软 AJAX 库的应用。

- 第7章“使用组件构建 AJAX 类库”，讲解使用微软 AJAX 库面向对象开发的 JavaScript 组件模型。
- 第8章“创建 AJAX 控件”，讲解使用微软 AJAX 库面向对象开发的 JavaScript 行为和控件模型。
- 第9章“AJAX and XSLT”，讲解 XSLT 基础知识以及如何使用 XSLT 在面向服务的 AJAX 应用中进行客户端渲染。
- 第10章“AJAX 和浏览器历史记录”，讲述使用微软 AJAX 库的客户端框架浏览历史维护能力。
- 第11章“使用面向对象的 AJAX 扩展 SharePoint”，本章涵盖了如何部署 Web 服务到 Windows SharePoint Services 3.0 和 Microsoft Office SharePoint Server 2007 上，以及如何在 Web PART 框架内部署面向服务的 AJAX 应用。

本书同时含有一个附录，“SharePoint 中的 WCF 支持”，一个在线文档，可以通过以下网址获得 <http://www.microsoft.com/mspress/companion/9780735625914>。

本书不涉及的内容

本书的创作意图是阐释面向服务的 AJAX 架构模式，而不是一本关于微软 AJAX 库、ASP.NET 和 WCF 的参考书。书中会充分介绍这些技术以便读者快速熟悉这种模式，使本书成为一本可读性强的参考书。读者可以参考 MSDN 相关文档或者使用自己喜爱的 Web 搜索引擎获取广泛的参考资料以便更深入地理解其中的概念。

因为我采用的是在面向服务的 AJAX 架构模式下的纯 AJAX 方式，所以本书中不会包含任何有关更新面板的文档。更新面板是一种伪 AJAX 方式，仅提供一些 AJAX 优点，主要还是利用类 AJAX 行为扩展已有代码。它不符合面向服务框架的架构模式，而且我也不推荐用它来做新的开发。关于更新面板技术有很多的参考资料，但是这本书不在其中。

书中也不会包含无障碍问题的相关讨论。这不是因为它不重要——其实很重要。坦率地讲，是因为没有时间而且在这方面我也不是专家。相反，我将这个问题留给读者，读者可以使用面向服务的 AJAX 模式来实现无障碍方案。既然这个架构模式能实现服务以提供数据流，

也能很简单地实现各种用户接口,不管是终端用户还是输入设备都很容易访问和使用,但是我会把这个问题留给无障碍专家来说明。

另外,书中没有太多的可用性提示,即使一个很重要的主题也不会有。我会将它留给读者,采用面向服务的 AJAX 架构模式来实现可用的应用。当用户在进行操作时,告诉用户系统正在处理他们的操作是一件非常重要的事情。如果加载数据需要一段时间,确保用户清楚他们所要的数据正在被加载而且应该给他们机会取消运行时间过长的任务。

最后,书中不会涉及任何测试版的技术的预览。现在有很多新技术值得期待,包括下一版本的 ASP.NET AJAX(可以通过 www.codeplex.com/aspnet 上的技术预览获得)和浏览器改进,比如 IE 8。还有未来浏览器中的跨页面的 XML HTTP 请求技术,可以允许 Web 服务调用远程的 Web 应用。而在本书中,Web 服务调用的是与 JavaScript 和 Web 页面在同一台服务器上的 Web 应用。

作者的博客

我的博客地址是 <http://daniellarson.spaces.live.com>。在博客上我会经常写一些关于 AJAX、SharePoint 技术文章和相关演讲以及作为一位父亲和一名软件开发工程师的生活文章。读者可以通过我的博客继续保持联系。

附属内容

本书的配套网站提供了书中所有使用的代码。代码是按照章节组织,读者可以通过以下地址登录配套网站下载相关代码: <http://www.microsoft.com/mspress/companion/9780735625914>。

软件与硬件需求

读者需要具备以下的软、硬件环境才能运行书中包含的相关配套内容。

- 微软 Windows Vista Home Premium 版, Windows Vista Business 版。
- Windows Vista Ultimate 版, Windows XP Professional, Windows 2003 Server Standard, Windows 2003 Server 企业版, Windows 2008 Server Standard, 或者 Windows 2008 Server 企业版 SP1。
- Microsoft Visual Studio 2008 标准版, Visual Studio 2008 企业版 SP1, 或者 Microsoft Visual C# 2008 Express 版和 Microsoft Visual Web Developer 2008 Express 版 SP1。
- 微软 .NET 3.5 SP1, 可以通过 <http://msdn.microsoft.com/vstudio/cc533448.aspx> 下载。
- 微软 SQL Server 2005 SP2; 微软 SQL Server 2005 Express 版 SP; 微软 SQL Server 2008 Express 版; 或者微软 SQL Server 2008。
- 1.6 GHz, 奔 III 及以上处理器。
- 1GB 内存。

另外,第 11 章“利用面向服务 AJAX 扩展 SharePoint”,需要基于微软 Windows Server 2008 或者微软 Windows Server 2003 上的 Window SharePoint Services 3.0。

本书基于 Windows Server 2008。

获取在线内容

对本书补充的新的或者更新的内容会被放到 Microsoft Press Online Developer Tools 网站上。读者可以找到包括更新本书内容、文章、配套内容的链接、勘误表、章节示例等资料。这些在线内容可以通过网站 <http://www.microsoft.com/learning/books/online/developer> 下载,网站上的内容会周期性更新。

本书相关支持

为了保证本书和相关配套内容的质量我们做了很多的努力。收集到的勘误和更新,会添加到微软知识库文章中。微软在以下网站提供书籍的相关支持:<http://www.microsoft.com/learning/support/books/>。

疑难解答

如果您对本书有意见、建议或者想法,或者是上述网站上未能解决的问题,请将它们通过以下的 E-mail 发送到微软出版社: mspinput@microsoft.com。

注意:上述地址不提供微软公司的软件产品支持。

关于作者

Daniel Larson 在开发社会性软件和企业级 AJAX 应用方面享有盛名。 he 现在是 NewsGator 公司(位于美国科罗拉多州首府丹佛)的企业开发团队的软件架构师。当他没和家人在一起时,你就会发现他说的写的都是 AJAX、企业开发或者是采用 AJAX 技术和微软企业级服务器的社会性软件相关的东西。他也是畅销的 SharePoint 开发书籍《Inside Microsoft Windows SharePoint Services 3.0》的作者之一。Daniel 现在和他美丽的妻子 Sallina 以及两个孩子一起生活在科罗拉多州。通过 <http://daniellarson.spaces.live.com> 可以联系 Daniel。

致谢

本书虽然是我一人所写，但事实上我背后有一个优秀的团队，他们和我一起提炼出了这些内容。我的审查团队是由非常出色的开发人员和微软 MVP 组成。此书的成功，离不开审查团队在内容上对我的巨大帮助。团队成员有 Mikhail Dikov、Darrin Bishop、Al Pascual、Alvin Bruney 以及 Morgan Everett。另外非常感谢 Mark Collins、Nick Swan、Todd Baginski 和 Sahil Malik，谢谢他们对各章节的反馈。我要感谢 Sahil Malik 在 WCF 和 SharePoint 方面的出色创作。我还要感谢以下几个在微软的朋友，April Spence、Lawrence Lliu、Paul Andrew 以及 Melissa Travers，谢谢他们持之以恒的支持和帮助。

我还要郑重地感谢 Ted Pattison，谢谢他让我与他们一起参与《Inside Windows SharePoint Services 3.0》一书的写作。这本书是《Inside Windows SharePoint Services 3.0》中 AJAX Web 部分的延续，没有那本书的铺垫，我是不可能完成本书的写作的。

我还要感谢 Ben Ryan，谢谢他促成此书的写作，尤其是在当前市场到处充斥着 AJAX 书籍的时候。我坚信，本书是独一无二的。

编辑方面，John Pierce 负责修饰。John，我很吃惊于你的编辑技巧以及为本书添加的可读性。Per Blomqvist 负责技术内容审校，对技术相关内容提出了宝贵的修改意见。代码能运行，我们得谢谢 Perl！额外感谢 Devon Musgrave，谢谢他帮我开始这本书并进行早期的相关编辑。我要感谢整个微软出版社，是你们让这一切变为现实——你们真真正正地让这次写作成为一个难得的经历，你们帮助我写出了一本好书。没有你们我不会写书！

编委会方面，我非常感谢像 Joe Mayo、Roy Ogborn 以及微软丹佛开发团队这样的朋友，特别感谢团队的领导者——知名人士 Chris Wallace，是他让我开始使用微软技术。当我刚来丹佛的时候，我刚学 .NET 1.0，丹佛团队的很多人对我的成功做出了很多贡献。特别感谢送给 Roy Haschenberger 和他的家人，谢谢他们多年来对我的支持。还有 Amos，谢谢你买我的书。

我要特别感谢 NewsGator 的企业开发团队，谢谢你们在过去的几年里帮助完善这一体系结构，谢谢你们能给我一个平台实现这些想法。尤其感谢企业开发团队中的 Lane Mohler、Sherstin Lauman、Brian Agnes、Josh Aragon 和 Tom McIntyre，以及 NewsGator 管理层的 Brian Kellne、Ashley Roach、Karyn German 和 Greg Reinacker，谢谢你们通过公司和社会网络支持我的这些成果。这本书很大程度上是关于我们在 NewsGator 一直开发和完善的架构模式的，即服务开发第一，AJAX 应用第二。同时我也想给 NewsGator 的其他 AJAX 开发者传递信息“那是回传吗？”。

最后，我要感谢我的妻子对我写作的支持。Sallina，你是一个非常了不起的妻子，没有你的支持，我是写不成这本书的。将本书献给我的两个孩子，他们将永远也不必写回传代码。

目 录

第 1 部分 AJAX 与 WCF

第 1 章 面向服务的 AJAX 基本原理3

- 1.1 基于服务的 AJAX: 基本架构3
- 1.2 客户端运行环境4
 - 1.2.1 ASP.NET AJAX ScriptManager 管理5
 - 1.2.2 页面生存期与 Sys. Application6
 - 1.2.3 使用 Sys.Net.WebRequest 完成 Web 请求8
 - 1.2.4 客户端展现10
- 1.3 WCF (Windows Communications Foundation) 简介13
 - 1.3.1 XML 终结点与 WCF14
 - 1.3.2 使用 WCF 更新数据17
- 1.4 总结22

第 2 章 AJAX 应用服务器端: 面向服务与 WCF23

- 2.1 WCF 作为 AJAX 服务平台的优势23
- 2.2 面向服务与基于服务的 AJAX 应用24
- 2.3 WCF 基础25
 - 2.3.1 WCF 协定26
 - 2.3.2 WCF 终结点28
 - 2.3.3 WCF 绑定30
 - 2.3.4 WCF 行为30

2.4 WCF 服务的 ASP.NET 运行库支持 33

2.5 案例学习: 知识库参考应用 34

- 2.5.1 基于协定的服务编程 35
- 2.5.2 通过 WCF 数据协定实现数据模式 37
- 2.5.3 ASP.NET 中 WCF 的安全与授权 42

2.6 使用 ASP.NET 2.0 ASMX 实现 AJAX Web 服务 42

2.7 本章小结 44

第 3 章 AJAX 应用服务器端: WCF 的 Web 编程模型 45

3.1 WCF 与 Web 编程模型 45

- 3.1.1 启用 Web 编程模型 46
- 3.1.2 从 WCF 访问的当前 Web 上下文环境 47
- 3.1.3 REST 架构形式和 WCF 48
- 3.1.4 使用 GET 方法获取数据 50
- 3.1.5 可读取的 URL 和 URI 模板 50
- 3.1.6 使用 UriTemplate 类构建 URL 52
- 3.1.7 使用 WebInvoke 来支持 REST 服务谓词 52

3.2 WCF 联合服务介绍 61

3.3 使用 ASP.NET 2.0 处理程序实现 REST Web 服务 66

3.4 本章小结 71

第2部分 AJAX 实践

第4章 采用微软 AJAX 库的 AJAX

运行程序.....75

4.1 深入理解客户端编程模型.....75

4.2 使用 ASP.NET 服务端组件 部署运行76

4.2.1 编译脚本资源78

4.2.2 (ScriptManager) 脚本管理器 编程模型79

4.3 面向对象的 JavaScript 基础.....83

4.4 Microsoft AJAX 库 JavaScript 类型扩展84

4.4.1 JavaScript Type 类.....85

4.4.2 JavaScript 命名空间.....85

4.4.3 类型与继承86

4.5 本章小结89

第5章 微软 AJAX 库90

5.1 Microsoft AJAX 库客户端 框架90

5.2 JavaScript 基本类型扩展91

5.2.1 数组扩展91

5.2.2 Error 扩展93

5.2.3 Sys 命名空间中的全局 类型与对象96

5.2.4 Sys.Net 中的网络库.....105

5.2.5 利用 Sys.Serialization 实现 JavaScript 序列化107

5.2.6 使用 Sys.Services 的应用 服务107

5.2.7 使用 Sys.UI 的浏览器 扩展107

5.3 Visual Studio 的智能感知 代码编写支持113

5.4 本章小结116

第6章 使用 Sys.Services 实现 AJAX

应用服务117

6.1 用 Sys.Services 实现 ASP.NET AJAX 应用服务118

6.1.1 ASP.NET AJAX 中的脚本初 始化与 Sys.Services119

6.1.2 通过 Sys.Services 实现 JavaScript 授权122

6.1.3 Sys.Services 管理 JavaScript 配置文件 (profile) 数据125

6.1.4 应用角色与 Sys. Services.RoleService127

6.2 构建在 WCF 之上的面向服务 的应用服务130

6.3 配置 WCF 应用服务136

6.4 本章小结138

第3部分 应用 AJAX

第7章 使用组件构建 AJAX 类库141

7.1 深入理解 AJAX 库中的属性 和事件142

7.1.1 属性142

7.1.2 事件143

7.2 使用 AJAX 库框架开发 组件147

7.2.1 将上下文传递给事件处理 程序151

7.2.2 构建应用服务组件153

7.3 本章小结165

第8章 创建 AJAX 控件166

8.1 AJAX 库 Behavior 类167

8.1.1 通过 AJAX 库实现自定义 行为	169	10.2 Microsoft AJAX 库中的 历史记录支持	223
8.1.2 Behavior 类静态方法	174	10.2.1 ScriptManager 控件的历史 记录支持	224
8.2 AJAX 库 Control 类	175	10.2.2 客户端 Sys.Application 对象 的历史记录支持	224
8.2.1 用 AJAX 库创建自定义 控件	178	10.3 案例学习: 给 Wiki 应用添加 历史记录功能	230
8.2.2 简单的确认和 property- Changed 事件	182	10.4 本章小结	232
8.3 案例学习: 创建一个 Wiki 应用控件	187	第 11 章 使用面向对象的 AJAX 扩展 SharePoint	233
8.4 本章小结	201	11.1 SharePoint 应用平台介绍	233
第 9 章 AJAX and XSLT	202	11.1.1 SharePoint 虚拟化 Web 应用	234
9.1 使用 XSLT 渲染 XML	202	11.1.2 SharePoint 的 Web 服务 选项	235
9.2 XSLT 基础	204	11.1.3 Web Part 应用程序	236
9.3 创建 XSLT 视图	206	11.2 通过 Web Parts 部署 代码	238
9.3.1 客户端 XML	209	11.3 应用实例: 创建在线 聊天室	253
9.3.2 构建 XML 控件	211	11.4 本章小结	270
9.4 案例学习: 使用 Xml 控件在 Wiki 应用中实现导航 功能	221		
9.5 本章小结	222		
第 10 章 AJAX 和浏览器历史记录	223		
10.1 历史记录	223		

第 1 部分

AJAX 与 WCF

本部分内容：

第 1 章 面向服务的 AJAX 基本原理	3
第 2 章 AJAX 应用服务器端：面向服务与 WCF	23
第 3 章 AJAX 应用服务器端：WCF 的 Web 编程模型	45

第 1 章

面向服务的 AJAX 基本原理

通过本章的学习，你将：

- 了解到面向服务的 AJAX 开发体系架构的基本原理。
- 理解客户端 JavaScript 运行环境中的 ScriptManager 控件以及 Microsoft AJAX 库的功能。
- 能够使用 ASP.NET AJAX 3.5 进行基本的 AJAX 编程。
- 能够使用 WCF 为 AJAX 实现简单的服务。
- 能够使用 AJAX 调用简单的 Web Service。

1.1 基于服务的 AJAX：基本架构

编写基于服务的 AJAX 应用，需要使用一种新的方法进行 Web 开发，并且在构思和构建应用程序方面进行根本性的转变。

在传统的 ASP.NET 应用中，创建的是基于页面的 Web Form 应用，我们称之为“经典 ASP.NET”。经典 ASP.NET Web Form 应用是单块的：使用经典的 ASP.NET 进行开发，需要为基于 Form 的用户界面创建功能。而在使用面向服务的 AJAX 应用时，功能不再捆绑到单个页面或者 Web Form，相反，它以可从任何页面或应用中调用的 Web Service 为基础。构成 AJAX 应用的 JavaScript 组件也不是与页面捆绑的，它们以组件的形式出现，页面与应用服务在需要时可以随时装载。



注意：

基于服务的 AJAX 应用是使用 JavaScript 组件与 Web Service 交互实现的。基于服务的 AJAX 是将页面服务的原则应用到 Web 应用开发中的。

使用 AJAX 编程可以选择多种架构，其中，面向服务的 AJAX 风格就非常适合，它具有最大限度的灵活性、重用性与可扩展性。Microsoft 依靠 WCF 在 Web Service 技术上的先进性以及 ASP.NET 的 AJAX 扩展中 JavaScript 对网络服务的支持，创建了一个功能强大的开发平台，不仅是面向服务的体系架构，也是基于服务的 AJAX 应用。

基于服务的开发模式可以将 API 层与 UI（用户界面）层清晰地分离开。要使用 Microsoft .NET 3.5 框架实现上述分离，首先要利用 WCF Web Service 技术实现 API，然后使用基于浏览器的客户端技术实现 AJAX 用户界面。客户端应用（用户界面）是采用基于 JavaScript 和动态 HTML 的 ASP.NET AJAX 扩展来构建的。基于服务的 AJAX 应用是客户端的运行环境，负责展现用户界面，调用后台服务。在这里，后台服务是可重用的、经过安全验证的以及面向服务的，它通过 WCF 终结点（Endpoint）