

图书基本信息

书名：<<ASP.NET 2.0 AJAX高级编程>>

13位ISBN编号：9787302179405

10位ISBN编号：7302179409

出版时间：2008-7

出版时间：清华大学出版社

作者：（美）吉布斯，（美）沃荷林 著，吴宏泉 译

页数：296

译者：吴宏泉

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>

内容概要

本书详细介绍了Microsoft AJAX Library的核心内容，并指导如何使用这个Microsoft的新架构把ASP.NET 2.0和JavaScript的面向对象的开发方式结合应用，构建具有更丰富UI功能、响应更快的Web应用程序。本书适用于有经验的、想要在应用程序中实现AJAX功能的ASP.NET开发人员，本书同样适用于想要结合使用ASP.NET和AJAX的经验丰富的Web开发人员。

作者简介

Matt Gibbs是Microsoft UI Framework and Services Team的开发经理，他领导的是一个天才的团队。此团队负责ASP.NET和AJAX Framework的开发，同时还负责IIS 7的Integrated Pipeline的开发。自从在1997年加入到IIS 4团队并使用“传统”的ASP以来，Matt一直致力于Microsoft的Web技术的开发，他还和别人合作撰写了几本有关ASP和ASP.NET的图书。

Dan Wahlin是Microsoft在ASP.NET和XML Web服务方面的MVP，他还是Interface Technical Training的.NET开发的讲师。

Dan是XML和ASP.NET开发人员网站（www.xmlforasp.net）的创始人，此网站致力于在Microsoft .NET平台上的XML、ADO.NET，和Web服务的开发和使用。

他还是INETA Speakers Bureau的一员，并且是几个讨论会的发言人。

Dan自己或和别人合作撰写了几本有关.NET的图书，包括ASP.NET 2.0 MVP Hacks和XML for ASP.NET Developers。

在编写代码和撰写文章或图书的闲暇时间，Dan喜欢创作和录制音乐，还喜欢和他的爱人和孩子一起打高尔夫和篮球。

Dan还会定期更新他在<http://weblogs.asp.net/dwahlin>和<http://blogs.interfaceit.com/dan-wahlins-blog>上的博客。

书籍目录

第1章 开发新一代Web应用程序 1.1 实现多功能Web应用程序 1.2 谁会从AJAX中受益 1.2.1 最终用户利益所在 1.2.2 开发人员利益所在 1.3 什么是ASP.NET AJAX 1.3.1 ASP.NET组件 1.3.2 JavaScript组件 1.3.3 Web服务组件 1.3.4 动态HTML组件 1.4 其他AJAX库 1.5 使用ASP.NET AJAX平衡客户端和服务器编程 1.6 本章小结第2章 局部网页更新 2.1 UpdatePanel控件 2.1.1 触发更新 2.1.2 从服务器调用更新方法 2.1.3 局部页面更新生命周期 2.1.4 UpdatePanel的复杂性 2.2 UpdateProgress控件 2.3 使用Timer控件自动更新 2.4 管理页面请求 2.4.1 请求生命周期 2.4.2 取消请求 2.4.3 检测错误 2.4.4 使用更新 2.5 AJAX和可访问性 2.6 本章小结第3章 ASP.NET开发人员的JavaScript 3.1 客户端的JavaScript 3.1.1 把JavaScript部署到浏览器 3.1.2 感知性能 3.1.3 脚本资源 3.2 JavaScript变量类型 3.3 语言不规则性 3.3.1 什么是相等性 3.3.2 Null、Undefined或其他类型 3.4 变量作用域 3.5 使用原型 3.6 关闭闭包 3.6.1 垃圾收集 3.6.2 DOM元素 3.6.3 避免内存泄漏 3.7 使用eval求值 3.8 处理异常 3.9 字符串操作 3.10 更好的JavaScript 3.10.1 减少脚本 3.10.2 使用缓存变量 3.11 本章小结第4章 ASP.NET AJAX Client Library 4.1 浏览器页面生命周期 4.2 使用类型系统 4.2.1 声明命名空间 4.2.2 创建类 4.2.3 使用接口 4.2.4 定义枚举和标记 4.3 基本类库 4.3.1 String类 4.3.2 日期和数字 4.3.3 数组 4.4 Intellisense支持 4.5 侦错和调试 4.5.1 调试跟踪控制台 4.5.2 生成错误 4.5.3 验证参数 4.6 浏览器对象 4.7 本章小结第5章 使用ScriptManager 5.1 无处不在的ScriptManager 5.1.1 添加脚本引用 5.1.2 设置ScriptMode 5.1.3 嵌入脚本资源 5.1.4 脚本本地化 5.1.5 脚本全球化 5.1.6 服务引用 5.2 ASP.NET应用程序服务 5.2.1 表单身份验证 5.2.2 配置服务 5.3 错误处理 5.3.1 定制错误处理 5.3.2 处理异步回送错误 5.3.3 异步超时 5.4 定制脚本位置 5.4.1 设置脚本路径 5.4.2 解析脚本引用 5.4.3 代理 5.4.4 脚本压缩 5.5 兼容性 5.6 本章小结第6章 ASP.NET AJAX联网功能 6.1 XMLHttpRequest对象 6.1.1 XMLHttpRequest安全性 6.1.2 对象属性和方法 6.2 WebRequest调用 6.2.1 设置HTTP动词 6.2.2 设置超时限制 6.2.3 添加定制头 6.2.4 传递额外数据 6.2.5 解析完整URL 6.3 管理Web请求 6.3.1 默认超时设置 6.3.2 全局Web请求处理 6.3.3 WebRequestExecutor 6.4 传递数据 6.4.1 串行化 6.4.2 JSON格式 6.4.3 JSON串行化 6.4.4 PageMethods 6.4.5 使用日期 6.4.6 回避串行化 6.4.7 配置JSON串行化功能 6.4.8 定制串行化 6.5 本章小结第7章 ASP.NET AJAX应用程序服务 7.1 用户身份验证 7.1.1 配置表单身份验证 7.1.2 定制身份验证 7.1.3 使用JavaScript进行身份验证 7.1.4 访问用户状态 7.1.5 注销 7.2 用户Profile数据 7.2.1 定义Profile数据 7.2.2 访问Profile属性 7.2.3 在浏览器中访问Profile数据 7.2.4 重载Profile属性 7.2.5 加载Profile属性 7.2.6 存储Profile数据 7.3 本章小结第8章 多功能AJAX Toolkit控件 8.1 控制页面布局 8.1.1 Accordion 8.1.2 AlwaysVisibleControlExtender 8.1.3 CollapsiblePanelExtender 8.1.4 ResizableControl 8.1.5 Tab 8.2 UI效果 8.2.1 动画 8.2.2 UpdatePanel动画 8.2.3 DropShadow 8.2.4 RoundedComers 8.3 弹出式控件 8.3.1 日历 8.3.2 ConfirmButton 8.3.3 HoverMenu 8.3.4 PopupControl 8.3.5 ModalPopup 8.3.6 CascadingDropDown 8.4 本章小结第9章 测试和调试ASP.NET AJAX应用程序 9.1 调试和发布脚本 9.2 使用Error类 9.3 使用Sys.Debug类 9.3.1 执行跟踪操作 9.3.2 执行断言操作 9.4 启用Internet Explorer的调试功能 9.5 使用Internet Explorer和Microsoft Script Debugger调试 9.6 使用Firefox和Firebug调试 9.7 浏览ASP.NET AJAX请求和响应消息 9.7.1 使用Fiddler检测消息 9.7.2 使用Web Development Helper检测消息 9.8 本章小结第10章 部署ASP.NET AJAX应用程序 10.1 安装ASP.NET AJAX 10.1.1 ASP.NET AJAX 10.1.2 ASP.NET AJAX Control Toolkit 10.1.3 ASP.NET Futures CTP 10.2 Web场部署 10.2.1 设置MachineKey 10.2.2 操作会话状态 10.3 使用不同平台 10.4 避免常见的配置问题 10.4.1 禁用调试 10.4.2 启用定制错误 10.4.3 禁用跟踪 10.4.4 设置部署模式 10.5 创建Web部署项目 10.6 部署之后 10.6.1 性能监控 10.6.2 控制AJAX特征 10.6.3 配置IIS7 10.7 使用脚本压缩 10.7.1 压缩动态脚本 10.7.2 压缩静态脚本 10.7.3 考虑使用共享脚本 10.7.4 不要改变版本路径 10.8 本章小结第11章 构建定制控件 11.1 构建客户端ASP.NET AJAX控件 11.1.1 扩展JavaScript 11.1.2 注册控件命名空间 11.1.3 创建控件构造函数 11.1.4 使用原型设计模式和JSON 11.1.5 定义控件属性 11.1.6 初始化控件和处理事件 11.1.7 定义控件方法 11.1.8 处置控件资源 11.1.9 注册定制控件类 11.1.10 创建客户端控件实例 11.2 构建服务器端ASP.NET AJAX控件 11.2.1 在服务器端控件中嵌入脚本 11.2.2 创建控件类和实现IScriptControl

11.2.3 重写Render和OnPreRender方法 11.2.4 在ASP.NET页面中使用定制ASP.NET AJAX控件 11.3 本章小结

章节摘录

插图：第1章 开发新一代Web应用程序Web应用程序一直不像桌面应用程序那样功能丰富及反应及时。

最终用户不必知道应用程序的工作细节，但是他们基本知道在浏览器中与一个网站交互的工作方式和使用安装在本地的桌面应用程序是完全不同的。

当开发队伍在设计一个新项目时，他们面对的第一个问题就是，最终用户会不会接受Web开发的局限性，或者他们是不是需要安装一个客户端的桌面应用程序。

Web应用程序可以从任何浏览器中访问，几乎是在任何地方都可以，但是通过在浏览器中运行的标记和脚本代码所提供的功能是有局限性的。

桌面应用程序也被称为“胖客户端”应用程序，它要求用户必须要在自己的电脑上进行安装，但是可以允许开发人员充分利用操作系统的高级鼠标和图形功能，但这些都Web浏览器中却很难实现，桌面应用程序还可以利用用户的电脑执行诸如离线存储等任务。

相反，Web应用程序可以通过修改服务器上运行的内容而更新，而且网站访问者可以瞬时得到最新的版本。

然而，要更新一个桌面应用程序就比较困难，因为不得不用户重新安装应用程序，或者构建应用程序包含一个智能的系统来实现自动更新。

Web应用程序使用的是一种零部署的模式，但是桌面应用程序则使用的是重型部署和配置模式。

桌面应用程序对比于浏览器应用程序一般可以提供一个更强功能的用户体验，但是，通过使用Web应用程序可以让用户在任何地方、使用任何操作系统来访问，而不用额外的工作。

另外，许多公司对员工在自己电脑上安装软件有严格的限制，而且通常员工没有安装一个新应用程序所需的管理权限，所以Web应用程序在很多情况下是唯一可行的选择。

1.1 实现多功能Web应用程序多年以前，如果拥有一个自己的网站，公司会觉得与众不同，但现在已经完全不是这样了。

目前，只有一个网站是远远不够的，现在的Web应用程序要具有直观地和客户动作交互的能力，并且能够预见用户的反应，这样的网站才是标新立异的。

本书介绍了ASENETAJAX如何解决Web开发的特有问题的，并且为自己的网站在用户体验上提高一个层次铺平了道路。

在本章中，我会介绍ASENETAJAX平台的主要组件，并介绍目前AJAX的一些其他选择。

编辑推荐

您已经为构建一个具有更丰富UI功能、响应更快的Web应用程序做好准备了吗？

《ASP.NET 2.0 Ajax高级编程》详细介绍了Microsoft AJAX Library，并指导如何使用这个Microsoft的新架构把ASP.NET 2.0和JavaScript的面向对象的开发方式结合应用。

通过《ASP.NET 2.0 Ajax高级编程》，您可以得到对ASP.NET AJAX体系结构的深层次理解，包括服务器控件、客户端JavaScript库，以及运行时环境。

《ASP.NET 2.0 Ajax高级编程》还提供了很多示例，可以发现这种新的技术给用户带来了许多方便之处。

通过这种技术构建的应用程序，可以操作浏览器的DOM模型，并且可以和服务器通信来更新用户正在浏览的内容，而不用等待刷新整个页面。

《ASP.NET 2.0 Ajax高级编程》的两位作者，一位是Microsoft ASP.NET AJAX开发队伍的经理，另一位是一个一直活跃在第一线的Microsoft MVP（Most Valuable Professional，最有价值专家）。

《ASP.NET 2.0 Ajax高级编程》全面介绍了Microsoft AJAX Library的核心内容，并介绍其中的AJAX控件。

《ASP.NET 2.0 Ajax高级编程》还提供了大量样例代码，从中可以学到如何充分利用备受关注的、共享资源的ASP.NET AJAX Toolkit控件，及其扩展。

《ASP.NET 2.0 Ajax高级编程》还会介绍如何在Visual Studio和浏览器中调试程序，并提供一些有效的测试方法。

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>