

<<ASP.NET 1.1高级编程>>

图书基本信息

书名：<<ASP.NET 1.1高级编程>>

13位ISBN编号：9787302096184

10位ISBN编号：730209618X

出版时间：2005-1-1

出版时间：清华大学出版社

作者：Dave Sussman,Alex Homer,李敏波

页数：1139

字数：1868000

译者：李敏波

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>

<<ASP.NET 1.1高级编程>>

内容概要

本书详细介绍了从零开始构建ASP.NET站点的完整过程。

五位专家级的作者利用他们丰富的ASP.NET编程经验，为广大读者展现了ASP.NET的强大功能与无穷魅力。

这本全面改版的图书提供了众多精选示例，所有的代码都是使用ASP.NET 1.1编写和测试的。

本书理论联系实际，全面深入地介绍了ASP.NET 1.1的强大功能和各种实用技术，内容包括ASP.NET Web页面的基础知识、服务器控件和数据管理，以及Web服务、程序调试、性能优化、程序迁移和实时应用程序等。

本书针对ASP.NET 1.1修正了所有代码和相关描述，同时介绍了1.1版本中对安全性和性能方面的改进。

通过本书的学习，读者可以全面掌握利用.NET Framework进行Web应用程序开发的技巧。

书籍目录

第1章 ASP.NET概述 1.1 是升级还是创新 1.2 开始使用ASP.NET 1.3 安装.NET 1.3.1 配置范例 1.3.2 运行范例 1.4 Visual Studio .NET 2003 1.5 在Visual Studio .NET 2003中创建ASP.NET应用程序 1.6 ASP.NET与ASP的区别 1.6.1 使用新技术的原因 1.6.2 ASP.NET的优点 1.7 编写ASP.NET页面 1.7.1 创建Web站点 1.7.2 页面范例 1.8 本章小结第2章 认识.NET Framework 2.1 .NET简介 2.1.1 .NET的组成 2.1.2 .NET战略 2.1.3 Windows DNA存在的问题 2.1.4 需要.NET的原因 2.2 公共语言运行库(CLR) 2.2.1 CLR服务 2.2.2 通用功能 2.2.3 使用对象 2.2.4 命名空间 2.2.5 通用类型系统 2.2.6 程序集——版本化和安全性代码 2.2.7 CLR和COM 2.2.8 中间语言 2.2.9 应用程序域 2.3 .NET Framework详解 2.3.1 应用程序开发技术 2.3.2 类库 2.3.3 基类库 2.4 ASP.NET的设计目标 2.4.1 消除对脚本引擎的依赖性 2.4.2 性能 2.4.3 在ASP基础上的升级 2.4.4 便于部署 2.4.5 强大的工具支持 2.4.6 更简单、更灵活的配置 2.4.7 分解的“开放”设计 2.4.8 与语言无关 2.4.9 代码更少、更简练、更易于维护 2.4.10 功能强大的身份确认模型 2.4.11 正确认识错误的存在 2.5 本章小结第3章 .NET语言 3.1 所提供的语言 3.1.1 VBScript何去何从 3.1.2 Visual Studio还是Notepad 3.2 Visual Basic .NET 3.2.1 面向对象功能 3.2.2 语言的变化 3.3 C# 3.3.1 类 3.3.2 方法 3.3.3 属性 3.3.4 构造函数 3.3.5 析构函数 3.3.6 继承性 3.3.7 接口 3.3.8 引用 3.3.9 异常处理 3.3.10 XML文档 3.3.11 非安全代码 3.3.12 运算符重载 3.4 JScript .NET 3.5 C++ 3.6 Visual J# .NET 3.7 所支持的其他语言 3.8 .NET语言编译器 3.9 CLR的优点 3.9.1 通用API 3.9.2 通用类型 3.9.3 跨语言继承性 3.9.4 跨语言调试和记录 3.10 性能问题 3.10.1 语言 3.10.2 后期绑定代码 3.11 常用范例 3.11.1 变量声明 3.11.2 函数和过程 3.11.3 语法的区别 3.11.4 循环 3.11.5 类型转换 3.12 本章小结第4章 编写ASP.NET页面 4.1 编码问题 4.1.1 以前的编码方式 4.1.2 ASP.NET页面中的编码 4.1.3 回送识别 4.2 Page类 4.2.1 HttpRequest对象 4.2.2 HttpResponse对象 4.2.3 页面处理步骤 4.2.4 页面状态 4.2.5 页面指令 4.3 后台编码 4.3.1 开发工具中的后台编码 4.3.2 使用后台编码 4.3.3 在ASP.NET页面中继承后台编码类文件 4.4 页面高速缓存 4.4.1 输出高速缓存范例 4.4.2 通过浏览器实现高速缓存 4.5 智能导航 4.6 用户控件 4.6.1 建立用户控件的方法 4.6.2 用户控件属性 4.6.3 用户控件事件 4.6.4 用户控件的后台编码 4.6.5 用户控件的局部页面高速缓存 4.7 本章小结第5章 服务器控件与验证控件 5.1 服务器控件的概念 5.1.1 再论ASP.NET页面模型 5.1.2 练习使用服务器控件 5.1.3 关于范例应用程序 5.1.4 使用服务器控件的场合 5.1.5 ASP.NET中的可用控件 5.2 HTML服务器控件 5.2.1 HtmlControl基类 5.2.2 HTMLGenericControl类 5.2.3 HTML控件的专有类 5.2.4 HTML服务器控件的用法 5.2.5 HTML服务器控件范例 5.3 ASP.NET输入验证控件 5.3.1 输入验证控件的作用 5.3.2 BaseValidator类 5.3.3 验证控件的特有成员 5.3.4 验证控件的用法 5.4 本章小结第6章 ASP.NET Web Form控件 6.1 基本的ASP.NET Web Form控件 6.1.1 为什么要有另外一组控件 6.1.2 基类WebControl 6.1.3 特殊的Web Form控件类 6.1.4 使用Web Form控件 6.1.5 ASP Web Form控件范例 6.2 ASP.NET列表控件 6.2.1 ListControl基类 6.2.2 特殊的列表控件类 6.2.3 使用ASP列表控件 6.3 ASP.NET中的其他多功能控件 6.4 本章小结第7章 列表控件和数据绑定 7.1 获取范例文件 7.2 数据绑定的相关概念 7.2.1 在服务器上完成所有操作 7.2.2 显示数据——ASP和ASP.NET 7.2.3 数据绑定的语法 7.2.4 单值数据绑定 7.2.5 重复值数据绑定 7.2.6 重复值数据绑定的简单范例 7.3 添加样式和模板 7.3.1 使用样式属性 7.3.2 给DataGrid控件添加样式 7.3.3 对数据绑定控件使用模板 7.3.4 处理数据绑定事件 7.3.5 对DataGrid中的行进行排序和筛选 7.3.6 DataGrid中的自动分页 7.4 使用数据绑定控件编辑数据 7.4.1 利用DataGrid控件编辑数据 7.4.2 利用DataList控件选择和编辑数据 7.5 本章小结第8章 .NET数据管理引论 8.1 数据存储和数据访问 8.1.1 转向分布式环境 8.1.2 XML——将来的数据格式 8.1.3 是否是另一项数据访问技术 8.1.4 .NET意味着断开数据的连接 8.1.5 .NET意味着XML数据 8.1.6 .NET意味着托管代码 8.1.7 .NET意味着新的编程模型 8.2 .NET中的数据管理 8.2.1 数据管理的系统命名空间 8.2.2 导入所需的命名空间 8.3 基本的ADO.NET类 8.3.1 ADO和ADO.NET的技术比较 8.3.2 Connection类 8.3.3 Command类 8.3.4 DataAdapter类 8.3.5 DataSet类 8.3.6 DataTable类 8.3.7 DataView类 8.3.8 DataReader类 8.3.9 DataReader和DataSet的选取 8.3.10 .NET中的关系数据提供程序 8.4 .NET中常见的数据访问任务 8.4.1 在自己的系统上创建范例 8.4.2 ASP.NET与IIS 6.0和Windows

Server 2003 8.4.3 使用DataReader对象 8.4.4 一个简单的DataSet范例 8.4.5 含有多个表的DataSet范例
8.4.6 返回DataSet对象的用户控件 8.5 .NET中的XML引论 8.5.1 基本的XML对象 8.5.2 XPathNavigator
类 8.5.3 XmlTextWriter类 8.5.4 XmlReader类 8.5.5 XmlValidatingReader类 8.5.6 XslTransform类 8.6
.NET中常见的XML任务 8.6.1 通过DOM的XML文档访问操作 8.6.2 使用XPathNavigator对象的XML文
档访问操作 8.6.3 搜索XML文档 8.6.4 XML TextWriter对象范例 8.6.5 XML TextReader对象范例
8.6.6 XSL Transform对象范例 8.7 选择一种数据存储的方法 8.7.1 关键在于访问和处理 8.7.2 查询的
新方法 8.7.3 传输协议是大势所趋 8.7.4 .NET是一套伟大的解决方案 8.8 本章小结第9章 处理关系数
据 9.1 获得范例文件 9.2 访问复杂数据 9.2.1 利用DataReader访问复杂数据 9.2.2 通过DataSet访问复
杂数据 9.2.3 访问和显示嵌套的数据 9.2.4 使用数据库的存储过程 9.3 处理DataTable对象 9.3.1 创建
和填充新DataTable的代码 9.3.2 在表中添加数据行 9.3.3 为DataSet添加表 9.3.4 管理约束和默认值
9.3.5 指定主键和外键 9.3.6 添加、修改、移去和删除行 9.3.7 处理DataTable事件 9.3.8 使用表和列
映射 9.4 数据的排序和筛选 9.4.1 DataTable对象中的排序和筛选 9.4.2 DataView对象中的排序和筛选
9.5 本章小结第10章 关系数据源的更新 10.1 获得范例文件 10.2 利用Command对象更新数据 10.2.1
使用Command对象和SQL语句 10.2.2 使用存储过程和Command对象 10.3 使用事务处理更新数据源
10.3.1 数据库事务 10.3.2 基于连接的事务处理 10.4 更新DataSet对象中的数据 10.4.1 深入
到DataAdapter对象内部 10.4.2 使用DataAdapter.Update方法 10.4.3 使用存储过程更新DataSet 10.4.4
DataAdapter中的更新事件 10.5 对并发数据更新的管理 10.5.1 避免出现并发错误 10.5.2 一个并发错
误范例 10.5.3 仅更新已改变的列 10.5.4 利用RowUpdated事件捕获错误 10.5.5 更新完成后查找错误
10.6 本章小结 第11章 .NET中的XML数据管理 11.1 获取范例文件 11.2 XML和DataSet对象 11.2.1
DataSet基于XML的方法 11.2.2 将数据从DataSet写入XML文件 11.2.3 将数据从XML文件读入DataSet
11.2.4 以字符串的形式从DataSet中得到XML 11.2.5 DataSet中的嵌套XML和关系数据 11.3 处
理XmlDataDocument对象 11.4 直接从SQL Server中读取XML数据 11.5 验证XML文档的有效性 11.6 创
建和编辑XML文档 11.7 使用XSL和XSLT转换 11.7.1 一个XSL转换的范例 11.7.2 用XmlResolver加载样
式表和模式 11.8 本章小结第12章 Web应用程序和global.asax 12.1 IIS Web根目录以及应用程序 12.1.1
创建IIS虚拟目录 12.1.2 把文件夹标记为应用程序 12.1.3 Windows Server 2003 12.2 ASP.NET Web应用
程序 12.2.1 注册组件 12.2.2 应用程序代码: global.asax 12.3 应用程序状态的管理 12.3.1 Session——
管理用户状态 12.3.2 Application——管理应用程序状态 12.3.3 Cache——管理瞬时状态 12.4 应用程
序事件 12.4.1 事件的语法和原型 12.4.2 支持的事件 12.4.3 事件范例 12.5 高级主题 12.5.1 静态变
量 12.5.2 为global.asax使用自己的基类 12.5.3 将文件扩展名映射到ASP.NET中 12.5.4 异步应用程序
事件 12.6 本章小结第13章 ASP.NET的配置 13.1 配置综述 13.1.1 ASP的配置 13.1.2 ASP.NET的配置
13.1.3 配置文件格式 13.2 常用的配置设置 13.2.1 通用配置设置 13.2.2 页面配置 13.3 跟踪 13.3.1 传
统的ASP跟踪 13.3.2 ASP.NET跟踪 13.3.3 查看跟踪输出 13.3.4 跟踪配置设置 13.4 定制错误 13.4.1
错误模式 13.4.2 总是显示ASP.NET错误页面 13.4.3 定制错误页面 13.4.4 配置IIS和ASP.NET来支持
相同的错误页面 13.5 身份验证和授权 13.5.1 身份验证 13.5.2 护照验证 13.5.3 授权 13.6 Web服务
13.6.1 国际化和编码 13.6.2 编译选项 13.6.3 执行身份的控制 13.6.4 用HTTP处理程序来扩
展ASP.NET 13.6.5 用HTTP模块扩展ASP.NET 13.6.6 配置ASP.工作进程 13.6.7 机器密钥 13.7 高级问
题 13.7.1 指定位置 13.7.2 设置路径 13.7.3 锁定配置设置 13.7.4 定制配置处理程序 13.8 本章小结
第14章 保障ASP.NET应用程序的安全 14.1 Windows 2000和IIS安全性概述 14.1.1 安全性需求 14.1.2
安全性概念 14.1.3 Windows的身份验证过程 14.1.4 在IIS中验证身份 14.1.5 在Windows中授权
14.1.6 在IIS中授权 14.2 ASP.NET安全概述 14.2.1 ASP.NET安全进程 14.2.2 ASP.NET模仿 14.2.3
Windows Server 2003和IIS 6.0 14.2.4 ASP.NET安全选项 14.3 程序设计的安全和个性化处理 14.3.1 角色
和身份概述 14.3.2 获取用户的身份和角色 14.4 ASP.NET安全范例 14.4.1 获取范例文件 14.4.2 在服
务器上建立范例 14.4.3 Windows身份验证范例 14.4.4 基于表单的身份验证范例 14.4.5 简单的个性
化范例 14.5 ASP.NET进程账户和信任级别 14.5.1 指定进程账户 14.5.2 指定信任级别 14.6 本章小结
第15章 处理集合和列表 15.1 处理集合和列表 15.2 System.Collections核心接口 15.2.1 IEnumerable
和IEnumerator接口 15.2.2 ICollection和IList接口 15.2.3 列表排序——IComparer和Comparable 15.2.4
ICollection接口 15.2.5 处理字典对象 15.3 System.Collections.Specialized命名空间 15.3.1 StringCollection

类 15.3.2 StringDictionary类 15.4 强制类型的集合 15.4.1 定义定制类型 15.4.2 创建集合类 15.4.3 使用集合类 15.4.4 DictionaryBase和CollectionBase类 15.4.5 可任意处理的枚举对象 15.5 本章小结 第16章 使用其他基类 16.1 目录和文件的处理 16.1.1 类的概述 16.1.2 DirectoryInfo类和Directory类 16.1.3 File类和FileInfo类 16.1.4 常见的目录和文件操作 16.2 正则表达式 16.2.1 模式匹配 16.2.2 正则表达式类 16.3 本章小结第17章 .NET组件 17.1 编写业务对象 17.1.1 创建对象 17.1.2 编译类 17.1.3 测试类 17.2 跨语言继承 17.2.1 跨语言继承范例 17.2.2 .NET中的事务处理 17.2.3 服务组件 17.3 部署组件 17.3.1 什么是程序集 17.3.2 程序集和版本问题 17.3.3 程序集清单 17.3.4 并排执行 17.4 本章小结第18章 建立ASP.NET服务器控件 18.1 编写简单的控件 18.1.1 创建一个C#控件 18.1.2 Visual Basic 中的控件开发 18.1.3 多步页面显示 18.1.4 组合控件 18.2 编写文本框控件 18.2.1 IPostBackDataHandler接口 18.2.2 引发事件的控件 18.2.3 使用视图状态 18.2.4 类型和视图状态 18.2.5 关于对象属性和UI模板的更多内容 18.2.6 DataBind和OnDataBinding 18.3 其他问题 18.3.1 访问ASP.NET内部对象 18.3.2 编写自适应控件 18.3.3 控件属性 18.3.4 定制控件构建器 18.4 用户控件 18.5 本章小结 第19章 公布Web服务 19.1 Web服务概况 19.1.1 常见问题 19.1.2 规范 19.2 构建ASP.NET Web服务 19.3 编写ASP.NET Web服务 19.3.1 内联 19.3.2 后台编码 19.3.3 WebService指令 19.3.4 应用程序代码 19.4 协议 19.4.1 HTTP-GET 19.4.2 HTTP-POST 19.4.3 SOAP 19.5 数据类型 19.6 WebMethod和WebService属性 19.6.1 WebMethod属性 19.6.2 WebService属性 19.7 设计ASP.NET Web服务 19.7.1 繁琐和简洁 19.7.2 无状态与有状态 19.7.3 高速缓存与静态 19.7.4 异步与同步 19.8 高级ASP.NET Web服务 19.8.1 控制并整理XML 19.8.2 修改Web服务帮助页 19.8.3 SOAP头 19.9 本章小结 第20章 使用Web服务 20.1 查找和描述Web服务 20.1.1 UDDI 20.1.2 Web服务描述语言 20.2 构建Web服务代理 20.2.1 Fibonacci代理 20.2.2 使用Visual Studio .NET 20.2.3 使用wsdl.exe 20.3 使用代理类 20.3.1 控制超时 20.3.2 设置URL 20.3.3 为访问Web服务器设置代理 20.3.4 维护状态 20.4 从Web站点到Web服务 20.4.1 编写WSDL 20.4.2 构建代理 20.4.3 使用ScreenScrape代理 20.5 设计选择 20.5.1 SOAP异常 20.5.2 使用SOAP头文件 20.6 Web服务的安全性 20.6.1 ASP.NET身份验证和授权 20.6.2 HTTPS加密和客户证书 20.7 高级主题 20.7.1 定制的身份验证和授权 20.7.2 SOAP Extensions 20.8 本章小结第21章 移动控件 21.1 无线Web概述 21.1.1 WAP 21.1.2 WML 21.1.3 设备的互操作性 21.2 移动控件 21.2.1 简单范例 21.2.2 查看生成的代码 21.2.3 移动控件窗体 21.2.4 控件引用 21.2.5 mobileLabel 21.2.6 mobileLink 21.2.7 mobileImage 21.2.8 mobileCommand 21.2.9 mobileTextBox 21.2.10 mobileList 21.2.11 mobileSelectionList 21.2.12 mobileObjectList 21.2.13 mobile PhoneCall 21.2.14 mobileCalendar 21.2.15 mobileTextView 21.2.16 mobileAdRotator 21.2.17 mobilePanel 21.2.18 验证控件 21.3 移动控件的高级主题 21.3.1 设置样式 21.3.2 使用设备的功能 21.3.3 模板 21.3.4 分页 21.4 本章小结和未来展望 21.4.1 移动Internet的未来发展 21.4.2 移动控件的未来发展第22章 跟踪、错误处理、调试和性能 22.1 跟踪 22.1.1 页面级跟踪 22.1.2 写跟踪信息 22.1.3 应用程序级跟踪 22.2 错误处理 22.2.1 结构化的异常处理 22.2.2 ASP.NET 错误处理 22.2.3 向管理员通知错误 22.3 调试 22.3.1 SDK调试器 22.3.2 在Visual Studio .NET中调试代码 22.4 测试与性能 22.4.1 实施 22.4.2 抽样 22.4.3 定制抽样 22.4.4 分析 22.5 本章小结第23章 迁移和互操作性 23.1 迁移、集成还是重写 23.1.1 迁移 23.1.2 集成 23.1.3 重写 23.1.4 不做任何改动 23.2 ASP.NET 23.2.1 准备工作 23.2.2 内部对象 23.2.3 页面的改变 23.3 语言的变化 23.4 互操作性 23.4.1 跨边界 23.4.2 在.NET中使用COM对象 23.4.3 在COM中使用.NET组件 23.4.4 API调用 23.5 本章小结 第24章 案例研究——IBuyAdventure.NET 24.1 应用程序概述 24.2 IBuyAdventure.NET (IBA.NET) 24.2.1 目标用户 24.2.2 伸缩性——Web解决方案平台 24.2.3 设计企业的伸缩性 24.3 业务对象和程序集 24.3.1 ProductsDB业务对象 24.3.2 程序集 24.4 IBuyAdventure .NET 数据库 24.4.1 Accounts表 24.4.2 Orders表 24.4.3 Products表 24.4.4 ShoppingCarts表 24.5 应用程序的用户界面 24.6 IBuyAdventure.NET中的用户控件 24.6.1 只有一个服务器端的form元素 24.6.2 用C#来编写用户控件和代码 24.6.3 特殊用户控件——Specials.ascx 24.6.4 类别用户控件——Categories.ascx 24.6.5 产品详细说明 24.6.6 购物车 24.6.7 显示购物车和改变订单 24.6.8 结账处理与安全性 24.6.9 结账处理 24.7 本章小结 附录A 公用的系统命名空间附录B 提高性能的技巧 B.1 管理视图状态 B.2 管理会话状态 B.3 使用输出高速缓存 B.4 仅在合适的位置使用服务器控件 B.5 使用DataReader, 而不是DataSet B.6 使

<<ASP.NET 1.1高级编程>>

用SQL TDS类访问数据 B.7 尽可能使用数据绑定 B.8 在Visual Basic中使用Option Explicit或Strict B.9 对组件使用早期绑定 B.10 避免ASP兼容性问题 B.11 记住新的请求和响应对象附录C ASP.NET 1.1中的变动 C.1 新的命名空间 C.2 System命名空间中的变化 C.3 System.Data命名空间中的变化 C.4 System.IO命名空间中的变化 C.5 System.Net命名空间中的变化 C.6 System.Web命名空间中的变化 C.7 Web服务类中的变化 C.8 System.Xml命名空间中的变化附录D 参考资料及其他信息

<<ASP.NET 1.1高级编程>>

媒体关注与评论

书评本书主要内容：
· ASP.NET和.NET Framework的基础概念；
· 使用服务器控件以及数据管理技术创建ASP.NET1.1页面；
· 开发、保护和配置Web应用程序；
· 在1.1版本中对安全性和性能方面的改进；
· Web服务和ASP.NET在移动领域的应用；
· 调试、性能优化、程序迁移和互操作性能；
· ASP.NET 1.1技术在实际环境中的应用。

本书读者对象：
· 具有ASP编程基础知识的程序员；
· 熟悉VB或基于C的语法（C++、Java或C#）的程序员；
· 希望利用.NET Framework开发复杂的ASP.NET 1.1应用程序的程序员；
· 希望ASP.NET 1.1的新技术进行全面深入了解的读者。

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>