

<<ASP.NET深入解析>>

图书基本信息

书名：<<ASP.NET深入解析>>

13位ISBN编号：9787030122537

10位ISBN编号：7030122534

出版时间：2004-2-1

出版时间：科学出版社

作者：苏红超

页数：400

字数：593000

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深入解析>>

内容概要

本书是一本比较深入探讨ASP . NET技术的中高级读物。

书中通过详尽的实际范例，深入讲解了ASP . NET技术的重点内容，结合ASP . NET技术探讨了有关面向对象、设计模式、架构设计和多层开发方面的技术知识，以便读者学完本书后能更好地利用ASP . NET技术设计、开发出更具竞争力的企业级软件产品。

本书通过3个部分循序渐进地讲解ASP . NET的重点核心技术，由浅入深的讲解方式会使读者更顺畅地学习ASP . NET高级技术。

本书适合于对ASP . NET技术感兴趣的中高级读者，以及希望采用ASP . NET技术构建企业级应用软件的开发人员。

本书适合于对ASP.NET技术感兴趣的中高级读者，以及希望采用ASP.NET技术构建企业级应用软件的开发人员。

<<ASP.NET深入解析>>

书籍目录

| | | |
|---|---|---|
| 第一篇 基础知识 | 第1章 Microsoft .NET和ASP .NET技术框架 | 1.1 .NET Framework技术架构 |
| 1.1.1 .NET体系结构 | 1.1.2 .NET Framework通用语言运行时 | 1.1.3 .NET Framework基础类库 |
| 1.1.4 .NET Framework的安装和设置 | 1.2 ASP .NET技术架构 | 1.3 XML Web Service和SOAP简介 |
| 1.2.1 ASP .NET技术平台纵览 | 1.2.2 ASP .NET技术框架简介 | 1.3.1 XML简介 |
| 1.2.2 ASP .NET技术框架简介 | 1.3 XML Web Service和SOAP简介 | 1.3.2 XML基本语法 |
| 1.3.1 XML简介 | 1.3.3 XML架构定义语言简介 | 1.3.4 XSL简介 |
| 1.3.2 XML基本语法 | 1.3.4 XSL简介 | 1.3.5 Web Service简介 |
| 1.3.3 XML架构定义语言简介 | 1.3.5 Web Service简介 | 1.3.6 SOAP简介 |
| 1.3.4 XSL简介 | 1.3.6 SOAP简介 | 小结 |
| 1.3.5 Web Service简介 | 第2章 ASP .NET开发环境配置 | 2.1 ASP .NET开发环境的安装 |
| 1.3.6 SOAP简介 | 2.1 ASP .NET开发环境的安装 | 2.1.1 ASP .NET运行环境的安装 |
| 小结 | 2.1.2 ASP .NET开发环境的安装 | 2.2 ASP .NET开发环境的配置 |
| 第2章 ASP .NET开发环境配置 | 2.2 ASP .NET开发环境的配置 | 2.2.1 ASP .NET开发环境的简单配置 |
| 2.1 ASP .NET开发环境的安装 | 2.2.1 ASP .NET开发环境的简单配置 | 2.2.2 Visual Studio .NET环境的简单配置 |
| 2.1.1 ASP .NET运行环境的安装 | 2.2.2 Visual Studio .NET环境的简单配置 | 第3章 面向对象知识 |
| 2.2 ASP .NET开发环境的配置 | 第3章 面向对象知识 | 3.1 面向对象基础知识 |
| 2.2.1 ASP .NET开发环境的简单配置 | 3.1 面向对象基础知识 | 3.1.1 面向对象软件开发简介 |
| 2.2.2 Visual Studio .NET环境的简单配置 | 3.1.1 面向对象软件开发简介 | 3.1.2 面向对象设计和编程的基本原则 |
| 第3章 面向对象知识 | 3.1.2 面向对象设计和编程的基本原则 | 3.1.3 面向对象的软件开发模式(需求.设计.编码.测试) |
| 3.1 面向对象基础知识 | 3.1.3 面向对象的软件开发模式(需求.设计.编码.测试) | 3.1.4 常用的软件系统建模工具 |
| 3.1.1 面向对象软件开发简介 | 3.1.4 常用的软件系统建模工具 | 3.2 设计模式简介 |
| 3.1.2 面向对象设计和编程的基本原则 | 3.2 设计模式简介 | 3.2.1 设计模式的基础知识 |
| 3.1.3 面向对象的软件开发模式(需求.设计.编码.测试) | 3.2.1 设计模式的基础知识 | 3.2.2 23种基本设计模式介绍 |
| 3.1.4 常用的软件系统建模工具 | 3.2.2 23种基本设计模式介绍 | 3.2.3 设计模式在架构设计中的应用 |
| 3.2 设计模式简介 | 3.2.3 设计模式在架构设计中的应用 | 3.2.4 .NET中的设计模式简介 |
| 3.2.1 设计模式的基础知识 | 3.2.4 .NET中的设计模式简介 | 小结 |
| 3.2.2 23种基本设计模式介绍 | 小结 | 第二篇 深入解析 |
| 3.2.3 设计模式在架构设计中的应用 | 第二篇 深入解析 | 第4章 ASP .NET Framework |
| 3.2.4 .NET中的设计模式简介 | 第4章 ASP .NET Framework | 4.1 一个HTTP请求的生命周期 |
| 小结 | 4.1 一个HTTP请求的生命周期 | 4.1.1 回顾ASP时代的HTTP请求处理过程 |
| 第二篇 深入解析 | 4.1.1 回顾ASP时代的HTTP请求处理过程 | 4.1.2 ASP .NET时代的HTTP请求处理方式 |
| 第4章 ASP .NET Framework | 4.1.2 ASP .NET时代的HTTP请求处理方式 | 4.1.3 深入ASP .NET Framework HTTP运行时 |
| 4.1 一个HTTP请求的生命周期 | 4.1.3 深入ASP .NET Framework HTTP运行时 | 4.2 HttpModule是如何工作的 |
| 4.1.1 回顾ASP时代的HTTP请求处理过程 | 4.2 HttpModule是如何工作的 | 4.2.1 HttpModule在ASP .NET Framework中的位置 |
| 4.1.2 ASP .NET时代的HTTP请求处理方式 | 4.2.1 HttpModule在ASP .NET Framework中的位置 | 4.2.2 构建自己的HttpModule |
| 4.1.3 深入ASP .NET Framework HTTP运行时 | 4.2.2 构建自己的HttpModule | 4.2.3 深入了解HttpModule |
| 4.2 HttpModule是如何工作的 | 4.2.3 深入了解HttpModule | 4.3 初次接触IHttpHandler |
| 4.2.1 HttpModule在ASP .NET Framework中的位置 | 4.3 初次接触IHttpHandler | 4.3.1 IHttpHandler是什么 |
| 4.2.2 构建自己的HttpModule | 4.3.1 IHttpHandler是什么 | 4.3.2 IHttpHandler如何处理HTTP请求 |
| 4.2.3 深入了解HttpModule | 4.3.2 IHttpHandler如何处理HTTP请求 | 4.4 IHttpHandler深入解析 |
| 4.3 初次接触IHttpHandler | 4.4 IHttpHandler深入解析 | 4.4.1 实现IHttpHandler接口 |
| 4.3.1 IHttpHandler是什么 | 4.4.1 实现IHttpHandler接口 | 4.4.2 实现IhttpHandler工厂 |
| 4.3.2 IHttpHandler如何处理HTTP请求 | 4.4.2 实现IhttpHandler工厂 | 4.5 深入ASP .NET事件模型机制 |
| 4.4 IHttpHandler深入解析 | 4.5 深入ASP .NET事件模型机制 | 4.5.1 ASP .NET事件模型初步认识 |
| 4.4.1 实现IHttpHandler接口 | 4.5.1 ASP .NET事件模型初步认识 | 4.5.2 ASP .NET的事件模型深入了解 |
| 4.4.2 实现IhttpHandler工厂 | 4.5.2 ASP .NET的事件模型深入了解 | 小结 |
| 4.5 深入ASP .NET事件模型机制 | 小结 | 第5章 ASP .NET数据存取技术 |
| 4.5.1 ASP .NET事件模型初步认识 | 第5章 ASP .NET数据存取技术 | 5.1 ADO .NET技术简介 |
| 4.5.2 ASP .NET的事件模型深入了解 | 5.1 ADO .NET技术简介 | 5.1.1 ADO .NET简介和数据访问技术的历史 |
| 5.1 ADO .NET技术简介 | 5.1.1 ADO .NET简介和数据访问技术的历史 | 5.1.2 ADO .NET的新特性 |
| 5.1.1 ADO .NET简介和数据访问技术的历史 | 5.1.2 ADO .NET的新特性 | 5.2 ADO .NET技术深入解析 |
| 5.1.2 ADO .NET的新特性 | 5.2 ADO .NET技术深入解析 | 5.2.1 深入了解DataSet.Data Reader以及数据存取技术 |
| 5.2 ADO .NET技术深入解析 | 5.2.1 深入了解DataSet.Data Reader以及数据存取技术 | 5.2.2 编写可移植的数据访问程序 |
| 5.2.1 深入了解DataSet.Data Reader以及数据存取技术 | 5.2.2 编写可移植的数据访问程序 | 5.2.3 深入了解ADO .NET的事务控制 |
| 5.2.2 编写可移植的数据访问程序 | 5.2.3 深入了解ADO .NET的事务控制 | 5.2.4 ADO .NET同XML的融合 |
| 5.2.3 深入了解ADO .NET的事务控制 | 5.2.4 ADO .NET同XML的融合 | 5.3 XML数据有取技术简介(XML DOM & XML SAX) |
| 5.2.4 ADO .NET同XML的融合 | 5.3 XML数据有取技术简介(XML DOM & XML SAX) | 5.3.1 XML DOM简介 |
| 5.3 XML数据有取技术简介(XML DOM & XML SAX) | 5.3.1 XML DOM简介 | 5.3.2 XML SAX简介 |
| 5.3.1 XML DOM简介 | 5.3.2 XML SAX简介 | 5.3.3 DOM技术和SAX技术的区别 |
| 5.3.2 XML SAX简介 | 5.3.3 DOM技术和SAX技术的区别 | 小结 |
| 5.3.3 DOM技术和SAX技术的区别 | 小结 | 第6章 ASP .NET Web Form |
| 小结 | 第6章 ASP .NET Web Form | 6.1 Web Form简介 |
| 第6章 ASP .NET Web Form | 6.1 Web Form简介 | 6.1.1 Web Form编程机制 |
| 6.1 Web Form简介 | 6.1.1 Web Form编程机制 | 6.1.2 Web Form同Win Form的区别 |
| 6.1.1 Web Form编程机制 | 6.1.2 Web Form同Win Form的区别 | 6.2 HTML Server Control和Web Server Control |
| 6.1.2 Web Form同Win Form的区别 | 6.2 HTML Server Control和Web Server Control | 6.2.1 HTML Server Control重要控件讲解 |
| 6.2 HTML Server Control和Web Server Control | 6.2.1 HTML Server Control重要控件讲解 | 6.2.2 Web Server Control重要控件讲解 |
| 6.2.1 HTML Server Control重要控件讲解 | 6.2.2 Web Server Control重要控件讲解 | 6.2.3 HTML Server Control和Web Server Control的对比 |
| 6.2.2 Web Server Control重要控件讲解 | 6.2.3 HTML Server Control和Web Server Control的对比 | 6.3 编写自己的User Control |
| 6.2.3 HTML Server Control和Web Server Control的对比 | 6.3 编写自己的User Control | 6.3.1 用户控件简介 |
| 6.3 编写自己的User Control | 6.3.1 用户控件简介 | 6.3.2 编写自己的简单用户控件 |
| 6.3.1 用户控件简介 | 6.3.2 编写自己的简单用户控件 | 6.3.3 DHTML技术简介 |
| 6.3.2 编写自己的简单用户控件 | 6.3.3 DHTML技术简介 | 6.3.4 结合DHTML编写用户控件 |
| 6.3.3 DHTML技术简介 | 6.3.4 结合DHTML编写用户控件 | 6.4 在Web Form中结合客户端脚本 |
| 6.3.4 结合DHTML编写用户控件 | 6.4 在Web Form中结合客户端脚本 | 6.4.1 客户端脚本在Web Form中的位置 |
| 6.4 在Web Form中结合客户端脚本 | 6.4.1 客户端脚本在Web Form中的位置 | 6.4.2 在Web Form中控制客户端脚本 |
| 6.4.1 客户端脚本在Web Form中的位置 | 6.4.2 在Web Form中控制客户端脚本 | 6.5 Web Form代码重用 |
| 6.4.2 在Web Form中控制客户端脚本 | 6.5 Web Form代码重用 | 6.5.1 代码绑定技术实现代码重用 |
| 6.5 Web Form代码重用 | 6.5.1 代码绑定技术实现代码重用 | 6.5.2 用户控件技术实现代码重用 |
| 6.5.1 代码绑定技术实现代码重用 | 6.5.2 用户控件技术实现代码重用 | 6.5.3 Code Behind技术和User Control技术的比较 |
| 6.5.2 用户控件技术实现代码重用 | 6.5.3 Code Behind技术和User Control技术的比较 | 小结 |
| 6.5.3 Code Behind技术和User Control技术的比较 | 小结 | 第7章 ASP .NET Web Service |
| 小结 | 第7章 ASP .NET Web Service | 7.1 Web Scrvice的前景 |
| 第7章 ASP .NET Web Service | 7.1 Web Scrvice的前景 | 7.2 Web Service在ASP .NET中的具体实现 |
| 7.1 Web Scrvice的前景 | 7.2 Web Service在ASP .NET中的具体实现 | 7.2.1 利用Visual Studio .NET构建Web Service |
| 7.2 Web Service在ASP .NET中的具体实现 | 7.2.1 利用Visual Studio .NET构建Web Service | 7.2.2 手工构建Web Service代理类 |
| 7.2.1 利用Visual Studio .NET构建Web Service | 7.2.2 手工构建Web Service代理类 | 7.2.3 实现ASP .NET下的Web Service应用范例 |
| 7.2.2 手工构建Web Service代理类 | 7.2.3 实现ASP .NET下的Web Service应用范例 | 7.3 深入Web Service中的SOAP和WSDL |
| 7.2.3 实现ASP .NET下的Web Service应用范例 | 7.3 深入Web Service中的SOAP和WSDL | 7.3.1 SOAP同Web Service的关系 |
| 7.3 深入Web Service中的SOAP和WSDL | 7.3.1 SOAP同Web Service的关系 | 7.3.2 自定义SOAP Header |
| 7.3.1 SOAP同Web Service的关系 | 7.3.2 自定义SOAP Header | 7.3.3 WSDL技术规范简介 |
| 7.3.2 自定义SOAP Header | 7.3.3 WSDL技术规范简介 | 7.4 深入Web Service中的状态和事务管理 |
| 7.3.3 WSDL技术规范简介 | 7.4 深入Web Service中的状态和事务管理 | 7.4.1 Web Service的状态控制 |
| 7.4 深入Web Service中的状态和事务管理 | 7.4.1 Web Service的状态控制 | 7.4.2 Web Service的分布式事务控制 |
| 7.4.1 Web Service的状态控制 | 7.4.2 Web Service的分布式事务控制 | 7.5 Web Services Enhancements技术 |
| 7.4.2 Web Service的分布式事务控制 | 7.5 Web Services Enhancements技术 | 7.6 使用WSE来传送一个非文本格式的文件 |
| 7.5 Web Services Enhancements技术 | 7.6 使用WSE来传送一个非文本格式的文件 | 小结 |
| 7.6 使用WSE来传送一个非文本格式的文件 | 小结 | 第8章 ASP .NET配置.安全和缓冲机制 |
| 小结 | 第8章 ASP .NET配置.安全和缓冲机制 | 8.1 Web. Config文件的配置 |
| 第8章 ASP .NET配置.安全和缓冲机制 | 8.1 Web. Config文件的配置 | 8.1.1 Web. Config文件的作用 |
| 8.1 Web. Config文件的配置 | 8.1.1 Web. Config文件的作用 | 8.1.2 Web. Config中重要小节的配置讲解 |
| 8.1.1 Web. Config文件的作用 | 8.1.2 Web. Config中重要小节的配置讲解 | 8.2 Machine. Config文件的配置 |
| 8.1.2 Web. Config中重要小节的配置讲解 | 8.2 Machine. Config文件的配置 | 8.2.1 Machine. Config文件简介 |
| 8.2 Machine. Config文件的配置 | 8.2.1 Machine. Config文件简介 | 8.2.2 Machine. |

<<ASP.NET深入解析>>

| | | |
|--|--|--------------------------|
| Config的配置介绍 | 8.2.3 Machine.Config文件和Web.Config文件的差别 | 8.3 ASP.NET |
| 页面配置标记 | 8.3.1 ASP.NET页面标记的用途 | 8.3.2 ASP.NET页面的重要标记讲解 |
| | 8.3.3 充分利用ASP.NET的页面配置标记 | 8.4 ASP.NET中的安全管理 |
| ASP.NET中的身份安全验证机制 | 8.4.2 ASP.NET中的安全配置 | 8.4.3 ASP.NET的权限控制机制 |
| 缓冲编程 | 8.5 ASP.NET缓冲机制 | 8.5.1 ASP.NET缓冲方式 |
| 小结 | 8.5.2 ASP.NET | |
| 第三篇 开发实战 | | |
| ASP.NET分布式事务控制 | 第9章 ASP.NET高级应用开发 | 9.1 |
| 服务 | 9.1.1 COM+事务控制简介 | 9.1.2 在.NET下使用COM+服务 |
| ASP.NET状态对象使用性能优化 | 9.1.3 COM+事务控制开发实例 | 9.2 ASP.NET应用程序性能提升 |
| 页面负载性能优化 | 9.2.1 ASP.NET应用程序性能提升 | 9.2.2 ASP.NET数据存取性能优化 |
| 录 | 9.2.3 ASP.NET | 9.2.4 Web应用程序性能测试工具的使用 |
| Active Directory在Windows 2000 / XP中的管理作用 | 9.3 ASP.NET开发活动目录 | 9.3.1 Active Directory简介 |
| | 9.3.2 Active Directory管理工具的使用 | 9.3.3 |
| 9.3.5 利用Active Directory控制IIS | 9.3.4 .NET中对Active Directory的支持 | |
| 9.4.1 .NET下程序集简介 | 9.4 ASP.NET中的私有程序集组件和全局程序集组件 | |
| 9.4.4 程序集组件的密钥开发方式 | 9.4.2 实现私有程序集组件 | 9.4.3 实现全局程序集组件 |
| Mobile Internet Toolkit开发包简介 | 9.5 ASP.NET下开发Mobile应用程序 | 9.5.1 |
| 和PDA的ASP.NET应用程序 | 9.5.2 Mobile Web Forms控件 | 9.5.3 建立支持Mobile |
| Speech SDK简介 | 9.6 ASP.NET下开发Web语音应用程序 | 9.6.1 .NET |
| Framework的应用系统架构设计 | 9.6.2 开发自己的Web语音应用程序 | 小结 |
| DNA架构 | 10.1 优秀的软件架构简介 | 第10章 基于.NET |
| 架构和J2EE架构的区别 | 10.1.1 经典的Windows | |
| 历史回顾 | 10.1.2 .NET平台下的推荐架构 | 10.1.3 J2EE架构简介 |
| | 10.1.4 .NET | |
| | 10.2 设计基于.NET的N层分布式架构系统 | 10.2.1 N层架构的 |
| | 10.2.2 .NET下N层分布式架构的经典范例剖析 | 小结 |
| | | 第11章 附录 |
| 11.1 C#简要语法参考 | 11.1.1 #基本数据类型 | 11.1.2 C#基本流程语句 |
| 11.2 ASP.NET学习资料索引(网站) | 11.3 设计模式.架构设计学习资料索引 | |

<<ASP.NET深入解析>>

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>