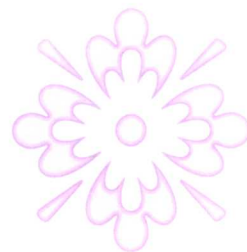
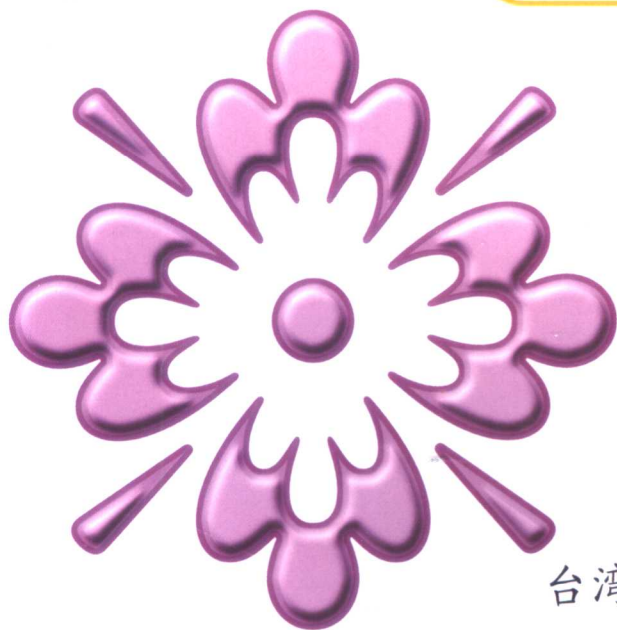


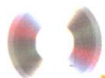
用实例学 ASP.NET

使用 C# 与 ADO.NET



台湾微软资深顾问 / 专业讲师

章立民 著



CD Included



電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

用实例学

ASP.NET

快速入门与进阶实例



ASP.NET 4.0 快速入门与进阶实例

王人虎 著

清华大学出版社

ISBN 7-302-25000-0

用实例学 ASP.NET **——使用 C#与 ADO.NET**

章立民 著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

这是一本能让读者真正深入到 ASP.NET 的参考书,为了让熟悉 C 语言的开发者能使用他们最熟悉的语言来开发 Web 应用程序,所以所有程序范例全部用 C#来撰写。本书讨论 ADO.NET 的篇幅虽然不多,但许多范例用到了 ADO.NET 的相关技术,以便让读者能熟悉 ADO.NET 的基本用法。本书范例众多,通过范例使学习深入是本书的鲜明特色。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

用实例学 ASP.NET: 使用 C#与 ADO.NET / 章立民著. —北京: 电子工业出版社, 2004.1
ISBN 7-5053-9205-0

I. 用… II. 章… III. ①主页制作—程序设计 ②C 语言—程序设计 ③数据库—接口—程序设计
IV. TP393.092

中国版本图书馆 CIP 数据核字 (2003) 第 088534 号

责任编辑: 周 筠

印 刷: 北京天竺颖华印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×980 1/16 印张: 40.25 字数: 842 千字

印 次: 2005 年 9 月第 4 次印刷

印 数: 2000 册 定价: 59.80 元(含光盘 1 张)

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。联系电话: (010) 68279077。质量投诉请发邮件至 zlt@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

序

笔者撰写本书的目的是希望众多相关领域的开发人员能有一本真正深入 ASP.NET 的参考书籍。放眼市面上, ASP.NET 的书籍不在少数, 不过大家切入的角度与观点却是大同小异, 太过初级是这些书籍的通病。事实上, ASP 早已被许多开发人员广泛使用, 他们具有一定的技术基础与经验, 而真正能打动这些读者的书籍是要立即切入核心并采用全新之开发观念的书籍, 本书正是以此理念来撰写的。

笔者深深以为能够使用 C# 程序语言来开发 ASP.NET 是一项重大突破, 因为它让 C 语言的开发者能够使用他们最熟悉的语言类型来开发 Web 应用程序, 因此, 在推出《用实例学 ASP.NET / 使用 VB.NET 与 ADO.NET》一书之后, 立即推出其姊妹作, 也就是本书《用实例学 ASP.NET / 使用 C# 与 ADO.NET》。虽然这两本书在架构上大同小异, 但是本书的所有程序范例全部是使用 C# 来撰写的, 期望能提供 C# 开发人员一个畅通的学习渠道。事实上, 笔者假设读者对 C# 语言有基本的素养, 因此, 不再浪费篇幅说明 C# 的基本语法与观念, 以期能毕其功于一役, 将所有的火力集中在 ASP.NET 本身。

此外, 虽然本书只花少许篇幅讨论 ADO.NET, 但是非常多的范例皆会使用到 ADO.NET 的相关技术。这种做法的好处是让读者能够在不知不觉中熟悉 ADO.NET 的基本用法, 以达到潜移默化之效。其实 ADO.NET 本身就是一个庞大的主题, 应该以专著来介绍, 更何况对一般的 ASP.NET 使用者来说, 仅使用到 ADO.NET 的基本功能。因此, 笔者认为本书的做法是一种最均衡的写法。

范例众多是本书的重要特色, 毕竟口沫横飞还不如一个实例来得清楚明白。虽然 ASP.NET 的推出已有一段时日, 然而, 近日参加微软公司的研讨会时, 竟发现许多开发人员仍然苦无良好的学习教材, 希望本书的推出能符合与满足大家的需求。

不可否认, Visual Studio .NET 确实是革命性的开发工具, 它的基础架构与开发环境更符合开发人员的理想, 此外, 其前瞻性与未来性更是令人耳目一新。相信在未来的日子里, 它将被更多的专业人员所采用。学会它, 肯定对您有所助益。

章立民, 笔于埔里

2002/08/24



作者小档案

姓 名： 章立民

出生地： 台湾省南投县埔里镇

性 别： 男

经 历：

- 台湾微软公司资深顾问与专业讲师
- 台湾工业研究院机械所制造资讯部顾问
- 台湾资诚会计师事务所资讯系统服务部顾问
- 台湾捷和建设资讯部顾问
- 台湾盘天科技顾问
- 台湾一日志工协会资讯顾问
- 《Computer DIY》杂志专栏执笔
- 《RUN!PC》杂志专栏执笔
- 曾任职于台湾空中大学、联电、药物食品检验局、调查局、南亚塑胶、NIKE、台核电一厂、大众电脑、台湾日立等台湾各大公民营企业与教学机构，乃台湾省内最权威之专业咨询顾问与讲师。

专 长： 关系数据库管理系统，目前致力于研究 ADO.NET、ASP.NET、Visual Basic.NET、Visual C#、SQL Server 2000、XML、Access 2002 与 FrontPage 2002。将有全系列书籍问世，敬请读者关注。

感 谢

本书之所以能够如期完成，要感谢周时超先生的大力协助，在此向他致上十二万分的谢意。

目 录

第 1 章 一窥究竟：ASP.NET 速览

1-1 ASP.NET 平台的系统需求	2
1-2 ASP.NET 与 ASP 的主要差异	2
结束语	7

第 2 章 ASP.NET 应用程序的灵魂：Web Form 网页

2-1 Web Form 网页的功能特性	10
2-2 Web Form 网页能够完成哪些操作	11
2-3 Web Form 网页的组件	13
2-4 Web Form 网页的代码模型	14
2-5 Web Form 网页的生存周期	17
2-6 Web Form 网页的处理阶段	20
2-7 ASP.NET 服务器控件事件模型	21
2-8 生成事件处理程序	30
2-9 在运行阶段绑定事件处理程序	32
2-10 将多个事件绑定至同一个事件处理程序	32
结束语	33

第 3 章 使用 Web 项目来生成与管理 Web Form 网页

3-1 为什么要使用 Visual Studio .NET	36
3-2 什么是 Web 项目	37
3-3 Web 项目文件的储存与访问	39
3-4 生成 Web 项目	41
3-5 打开现有的 Web 项目	43
3-6 以脱机方式使用 Web 项目	44
3-7 生成 Web Form 网页	45
3-8 生成事件处理程序	49
3-9 编译与执行 Web Form 网页	52



结束语.....	54
----------	----

第 4 章 万丈高楼平地起：Web Form 基本的语法与技巧

4-1 代码呈现块语法.....	56
4-2 代码声明块语法.....	58
4-3 服务器端注释语法.....	61
4-4 服务器端对象标记语法.....	62
4-5 服务器端 Include 指令语法.....	63
4-6 如何将用户重定向至其他网页.....	66
4-7 如何在 Web Form 网页中检测浏览器的类型.....	68
结束语.....	69

第 5 章 网页的基本元素：ASP.NET 服务器控件

5-1 ASP.NET 服务器控件的类型.....	72
5-2 使用建议事项.....	78
5-3 浏览器功能所影响的层面.....	79
5-4 如何将 Web 服务器控件加至 Web Form 网页.....	82
5-5 如何将 HTML 服务器控件加至 Web Form 网页.....	85
5-6 将 HTML 服务器控件转换回 HTML 项.....	86
5-7 以程序控制方式将控件加至 Web Form 网页.....	87
5-8 在设计阶段设置控件的属性.....	90
5-9 以程序控制方式设置 Web 服务器控件的属性.....	90
5-10 以程序控制方式设置 HTML 服务器控件的属性.....	96
5-11 设置服务器控件是否提交至服务器.....	98
5-12 ASP.NET 服务器控件与样式表.....	103
5-13 客户端脚本的影响层面.....	110
5-14 服务器控件的类成员数据.....	111
5-15 Label Web 服务器控件.....	112
5-16 Literal Web 服务器控件.....	115
5-17 TextBox Web 服务器控件.....	116
5-18 CheckBox Web 服务器控件.....	124
5-19 CheckBoxList Web 服务器控件.....	127
5-20 RadioButton Web 服务器控件.....	139
5-21 RadioButtonList Web 服务器控件.....	142

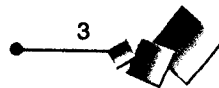




5-22 Button Web 服务器控件	152
5-23 ImageButton Web 服务器控件	160
5-24 LinkButton Web 服务器控件	163
5-25 DropDownList Web 服务器控件	164
5-26 ListBox Web 服务器控件	170
5-27 HyperLink Web 服务器控件	174
5-28 Image Web 服务器控件	177
5-29 AdRotator Web 服务器控件	178
5-30 Calendar Web 服务器控件	185
结束语	203

第 6 章 数据绑定与数据访问服务器控件 (DataGrid、DataList、Repeater 与 XML)

6-1 数据访问的基本概念	206
6-2 Web Form 网页的数据源	208
6-3 数据集、数据适配器与数据读取器	209
6-4 SQL Server .NET 数据提供者 vs. OLE DB .NET 数据提供者	211
6-5 数据访问策略	213
6-6 绑定属性	216
6-7 数据绑定表达式	216
6-8 使用 DataBinder 类进行绑定	228
6-9 数据绑定的时机与类型	230
6-10 数据绑定多条数据的 Web 服务器控件	231
6-11 如何在设计阶段绑定属性	233
6-12 如何在运行阶段绑定属性	239
6-13 活用 DataGrid Web 服务器控件	242
6-14 活用 DataList Web 服务器控件	312
6-15 活用 Repeater Web 服务器控件	346
6-16 活用 XML Web 服务器控件	352
6-17 数据对象与界面工具查询实际演练一	355
6-18 数据对象与界面工具查询实际演练二	364
6-19 数据对象与界面工具更新实际演练一	368
6-20 数据对象与界面工具更新实际演练二	375





结束语.....	384
----------	-----

第 7 章 数据检验的利器：验证控件

7-1 验证操作的基本逻辑与观念.....	386
7-2 验证控件的类型.....	387
7-3 客户端验证.....	388
7-4 特殊案例的验证结果.....	390
7-5 务必输入数据验证.....	391
7-6 匹配特定值验证.....	397
7-7 数据类型验证.....	401
7-8 格式验证.....	402
7-9 数据范围验证.....	404
7-10 执行自定义验证.....	406
7-11 测试验证控件的验证状态.....	416
7-12 自定义验证错误信息的显示方式.....	418
7-13 如何停用验证.....	425
7-14 以程序控制方式验证 ASP.NET 服务器控件.....	426
结束语.....	430

第 8 章 既有资源的快速集成者：用户控件

8-1 建立用户控件.....	432
8-2 将用户控件加至 Web Form 网页.....	433
8-3 以程序控制方式访问用户控件的属性.....	437
8-4 将 Web Form 网页转换为用户控件.....	446
结束语.....	448

第 9 章 来去一瞬间：谈 ASP.NET 的状态管理

9-1 以客户端为基础 vs. 以服务器端为基础.....	450
9-2 客户端状态管理功能：视图状态.....	452
9-3 客户端状态管理功能：隐藏窗体字段.....	461
9-4 客户端状态管理功能：Cookie.....	463
9-5 客户端状态管理功能：查询字符串.....	471
9-6 服务器端状态管理功能：应用程序状态.....	474



9-7 服务器端状态管理功能: 会话状态	479
9-8 服务器端状态管理功能: 数据库	484
9-9 如何传递服务器控件所持有的数据	485
结束语	491

第 10 章 应用程序的关键: Global.asax 文件

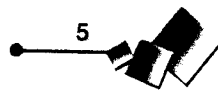
10-1 Global.asax 文件	494
10-2 HttpApplication 实例与事件处理	494
10-3 重写 Init 与 Dispose 方法	496
10-4 如何撰写事件处理程序	496
结束语	498

第 11 章 效能提升的催化剂: ASP.NET 缓存

11-1 缓存的类型	500
11-2 网页输出缓存	500
11-3 设置网页输出缓存的持续时间	501
11-4 设置网页输出缓存的缓存能力	503
11-5 缓存网页的各个版本	505
11-6 网页片段缓存	512
11-7 缓存用户控件的多个版本	515
11-8 应用程序数据缓存	518
11-9 将数据项加至缓存中	520
11-10 删除缓存中的数据项	527
结束语	537

第 12 章 让一切更完美: ASP.NET 配置设置

12-1 ASP.NET 配置系统的特性	540
12-2 配置文件的继承顺序	541
12-3 ASP.NET 配置文件的格式	544
12-4 标准的 ASP.NET 配置节	549
12-5 善用位置与路径	553
12-6 如何锁定配置设置	554
12-7 如何检索配置	555





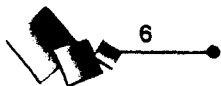
结束语.....	557
----------	-----

第 13 章 非请勿入：ASP.NET 安全性

13-1 Windows 2000 与 IIS 的安全性简介.....	560
13-2 ASP.NET 安全性处理流程.....	572
13-3 ASP.NET 用户账户模拟.....	573
13-4 ASP.NET 安全性配置设置.....	576
13-5 ASP.NET 验证.....	577
13-6 如何在 ASP.NET 中使用 Windows 验证.....	578
13-7 Windows 验证模式的程序设计技巧.....	586
13-8 如何在 ASP.NET 中使用窗体验证.....	590
13-9 窗体验证实现范例一.....	600
13-10 窗体验证实现范例二.....	607
13-11 窗体验证实现范例三.....	608
13-12 利用 XML 用户档案完成窗体验证.....	609
13-13 利用 SQL Server 数据库完成窗体验证.....	621
结束语.....	623

附录 范例安装与使用说明

A-1 安装范例文件.....	626
A-2 使用范例文件.....	628



一窥究竟： ASP.NET 速览



- ASP.NET 平台的系统需求
- ASP.NET 与 ASP 的主要差异

ASP.NET 与 ASP 的主要差异

如果说 ASP.NET 仅仅是 ASP 的升级版，那将是一个巨大的误解。从本质上讲，ASP.NET 与 ASP 是完全不同的两种技术。ASP.NET 是一个以 Common Language Runtime 为基础的、面向对象的、支持多语言的平台。它提供了比 ASP 更强大的功能，包括对数据库的连接、对 XML 的支持、对 Web 服务的访问等。ASP.NET 的架构更加清晰，代码更加规范，易于维护和扩展。此外，ASP.NET 还支持多种编程语言，如 C#、VB.NET 等，这为开发者提供了更多的选择。总的来说，ASP.NET 是一个更加先进、更加强大的 Web 开发平台，它代表了 Web 开发技术的未来发展方向。

多语言支持

正如大家所知道的，ASP 仅支持使用 Script 语言编写，而 ASP.NET 则支持多种语言，如 C#、VB.NET 等。这使得 ASP.NET 具有更强的灵活性和可扩展性。此外，ASP.NET 还支持对 XML 的支持，这使得它能够更方便地与各种数据源进行交互。总的来说，ASP.NET 是一个更加先进、更加强大的 Web 开发平台，它代表了 Web 开发技术的未来发展方向。

ASP.NET 是一个统一的 Web 开发平台，它提供生成企业级 Web 应用程序所需的各种服务。虽然 ASP.NET 有很大一部分的语法与 ASP (Active Server Pages) 兼容，但是它更提供了一个新的程序设计模型与架构，以便让您生成功能更强大且完善的应用程序。ASP.NET 完全在 .NET Framework 的支持之下，使得我们能够善加利用 Common Language Runtime (CLR)、类型安全性、继承性以及该平台的其他各项特性。本章将带领大家速览 ASP.NET，以便让诸位读者先有一个基本的认识与观念。

1-1 ASP.NET 平台的系统需求

ASP.NET 的服务器平台必须是 Windows 2000 或是已安装 Service Pack 6a 的 Windows NT 4.0。至于 Web Services，则在 Microsoft .NET Framework SDK 所支持的平台包括：Windows 2000、已安装 Service Pack 6a 的 Windows NT 4.0、Windows ME、Windows 98、Windows SE 上都可以使用，但是 Windows 95 除外。

1-2 ASP.NET 与 ASP 的主要差异

如果说 ASP.NET 仅仅是 ASP 的新版本，那不但过于笼统，也太小看 ASP.NET 了。事实上，ASP.NET 已经彻头彻尾地重新改造，它采用了全新的观念与服务器端技术来开发动态网页，足以适应未来数年或更长时间的开发需求。

ASP.NET 是一个以 Common Language Runtime 为基础所生成的程序设计架构，并且用来在服务器上生成功能强大的应用程序。ASP.NET 提供许多比过去的 Web 开发模型更加优异的功能，以下我们将详述其各项重要特性，而这些特性不仅代表它与 ASP 的主要差异，您也可以从中体会 ASP.NET 的精妙之处，进而了解为什么要从 ASP 转移至 ASP.NET。

▶▶ 多语言支持

正如大家所知道的，ASP 仅能使用 Script 语言编写服务器端脚本，最常用的不外乎是 VBScript 与 JavaScript。欲使用其他的 Script 语言，您的电脑必须安装其解析器。但是无论如何，Script 语言始终是解析式的，解析式语言的两大缺点就是缺乏严谨的类型（例如像 Visual Basic 或 Visual C / C++ 所支持的类型语言）与不具备编译环境。虽然

ASP 会缓存代码，但它仍然采用解析方式，因而不可避免地会发生效率低下与延展性不足的问题。

ASP.NET 是一个编译式的 .NET 环境，这意味着您不仅可以使⽤任何 .NET Framework 兼容的程序语言来编写 ASP.NET，还可以充分利⽤ .NET Framework 的 Common Language Runtime (CLR)、类型安全性、继承性与其他各项特性。显然您现在可以使⽤ .NET Framework 下的 Visual Basic .NET、Visual C# 或 JScript .NET 来编写 ASP.NET 而开发出更快速且可靠的动态网页。当然，您无须了解每一种程序语言，只须挑选您最熟悉的程序语言。比方说，如果您非常熟悉 Visual Basic .NET，就应该使⽤ Visual Basic .NET 来开发 ASP.NET；假如您对 Visual C# 情有独钟，当然也可以使⽤ Visual C# 来开发 ASP.NET。

▶▶ 更快的运行效率

ASP.NET 不仅能够享受到 .NET Framework 运行时在效率上的诸多强化，它本身亦经重新设计，使其运行效率大幅超越 ASP 与其他 Web 开发平台。所有的 ASP.NET 代码皆以编译过的公共语言运行时代码运行于服务器上，而并非采用解析方式，此举使得 ASP.NET 能够使⽤早期绑定、严谨类型化、实时 (JIT) 编译、原始编译与缓存服务来大幅提升其运行效率。

ASP.NET 也很容易加以拆解，亦即开发人员能够轻易将与他们所开发的应用程序无关的模块（例如：一个 **session** 模块）加以移除。ASP.NET 内附 Performance Counters，开发人员与系统管理员可以使⽤它来监控与测试应用程序。

缓存是 ASP.NET 非常重要的一项特性，并且能够大大帮助您生成高效能的 Web 应用程序。事实上，要生成高效能且延展性佳的 Web 应用程序的重要因素之一，就是要能够将相关元素在它们被取⽤后随即储存在内存中，即您可以将这些元素储存在 Web 服务器本身或是取⽤通道中（例如：Proxy 服务器或浏览器）。这样一来，当一个访问请求与先前相同时，可避免再次生成信息而提升效率，当访问请求需要耗⽤大量的处理器时间与其他服务器资源时，效率的提升更是明显。利⽤缓存技术，您可以将网页输出或跨越 HTTP 的应用程序数据访问请求储存起来，以后再次使⽤时，服务器将不需再次生成它们，因此，节省了时间与资源。

比方说，假设您的网页会去查询数据库并返回大量数据，并且需要花费 15 秒钟才能完成显示操作。在 ASP.NET 中，您可以加入以下的网页指令：

```
<%@ OutputCache Duration="60" %>
```

以上的 @ OutputCache 网页指令会使查询结果被缓存 60 秒钟，这样一来，当用户刷新网页或另外一位用户探访它时，将会直接使⽤缓存中的网页而不再次查询数据



库。

ASP.NET 提供下列两种类型的缓存：

☐ **输出缓存**

输出缓存允许您将访问请求所产生的动态网页与用户控制响应储存起来。这样一来，如果后续的访问请求与先前相同，将可直接使用缓存中的数据，而不需要再次动态运行网页或用户控制代码。

☐ **传统的应用程序缓存**

借助此种缓存，您能够以程序控制方式将任意的对象储存至服务器的内存中，这样，您的应用程序便可省下重新生成这些对象的时间与资源。

▶▶ 世界级的工具支持

您可以在 Visual Studio 集成开发环境中使用丰富的工具箱与设计器来开发 ASP.NET，比方说，您可以使用具备“所见即所得”的编辑功能的 HTML 编辑器，将服务器控件拖放至网页，自动化开发等等。此外，您可以将自行开发的组件加入“工具箱”，使其在取用上与一般标准内置的控件完全相同。凡此种种，都将使您的开发效率大幅提升。

▶▶ 程序设计模型

当生成 ASP.NET 应用程序时，开发人员可以选用下列两种程序设计模型之一，或是将二者综合运用。

- ☐ **Web Forms** 允许您生成功能强大的窗体网页。当生成此类网页时，您可以使用 ASP.NET 服务器控件去生成公用的用户接口元素，并替它们编写程序来完成一般性工作。这些控件允许您利用可重复使用的内置或自定义控件来反复生成一个 Web 窗体，如此便能简化网页的代码。
- ☐ **Web 服务** 是一种远程访问服务器功能的方法，它能够将数据或商业逻辑以程序化接口来呈现，以便让客户端或服务器应用程序来访问。Web 服务使用 HTTP 与 XML 信息传递标准来跨越防火墙传递数据，打开了“**客户端对服务器**”或“**服务器对服务器**”的数据交换模式。Web 服务并未依附于特定的组件技术或对象调用协议，也就是说，您所编写用来访问 Web 服务的程序能够使用任何的程序语言，使用任何的对象模型，并运行于任何的操作系统上。

上述两种做法皆能够善用 ASP.NET 的所有特性，以及 .NET Framework 及其 Common Language Runtime 的所有功能。