

PROGRAMMER TO PROGRAMMER™



ASP.NET Website Programming
Problem - Design - Solution
Visual Basic .NET Edition

ASP.NET Web 站点 高级编程

提出问题 - 设计方案 - 解决方案

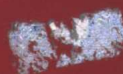
—— Visual Basic .NET 版本

Marco Bellinaso

Kevin Hoffman

杨 浩

著
译



清华大学出版社

ASP.NET Web 站点高级编程

提出问题-设计方案-解决方案

—— Visual Basic .NET 版本

Marco Bellinaso

Kevin Hoffman

著

杨 浩

译

清华大学出版社

北 京

北京市版权局著作权合同登记号：01-2002-6523

内 容 简 介

本书通过开发一个完整实用的 Web 站点实例，系统全面地介绍了 ASP.NET Web 站点开发技术。全书共分 12 章，各章都遵循提出问题、设计方案和解决方案的顺序。首先提出需要解决的问题，然后说明实现的过程，最后在 ASP.NET 中编写具体的软件代码。每一章都是独立的，分别介绍了如何开发一个可扩展的可交换的模块。这些章中大部分都涉及到创建 3 层的系统，包括数据层、业务层和表示层。书中还推荐了一个基本的站点构建框架，您可以利用该框架方便快捷地创建一些模块并把它们插入到 Web 站点中。

本书适用于 ASP.NET 编程的 Web 站点开发人员。

EISBN: 1-86100-816-3

Marco Bellinaso, Kevin Hoffman: ASP.NET Website Programming Problem-Design-Solution Visual Basic .NET Edition

Copyright© 2003 by Wrox Press Ltd.

Authorized translation from the English language edition published by Wrox Press Ltd.

All rights reserved.

本书中文简体字版由英国乐思出版公司授权清华大学出版社出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，翻印必究。

本书封面贴有清华大学出版社激光防伪标签，无标签者不得销售。

图书在版编目(CIP)数据

ASP.NET Web 站点高级编程：提出问题—设计方案—解决方案：Visual Basic .NET 版本 / (美)贝尼纳索，(美)霍夫曼著；杨浩译. — 北京：清华大学出版社，2003

书名原文：Marco Bellinaso, Kevin Hoffman: ASP.NET Website Programming Problem-Design-Solution Visual Basic .NET Edition

ISBN 7-302-06536-5

I. A... II. ①贝... ②霍... ③杨... III. ①主页制作 - 程序设计②BASIC - 程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字(2003)第 026703 号

出 版 者：清华大学出版社(北京清华大学学研大厦，邮编 100084)

[http:// www.tup.com.cn](http://www.tup.com.cn)

责任编辑：徐燕萍

印 刷 者：北京通州大中印刷厂

发 行 者：新华书店总店北京发行所

开 本：787×1092 1/16 印张：31.25 字数：800 千字

版 次：2003 年 4 月第 1 版 2003 年 4 月第 1 次印刷

书 号：ISBN 7-302-06536-5/TP·4903

印 数：0001~4000

定 价：65.00 元

前 言

欢迎阅读本书。本书将使用可扩展的、可交换的模块创建一个交互式的、基于内容的 Web 站点。学习完本书，您就可以掌握足够的 ASP.NET 编程技巧来开发高效的、设计良好的、可扩展的 Web 站点。

ASP.NET 是一个用于创建 Web 站点的优秀工具。它包含许多采用数千行传统 ASP 代码实现的内置功能。它还不需要管理员权限来部署编译的组件，整个站点都可以部署在一个文件夹中。

本书将详细介绍 ASP.NET 开发人员可以使用的各种功能，并着重介绍一些最有用的功能。

本书的重点是基于内容的 Web 站点。我们将不介绍如何创建电子商务系统，但书中的许多建议同样适用于电子商务站点。例如，可以使用相同的构造基础添加一个购物篮模块。

本书与 Wrox 公司的其他书都不一样，因为我们将用整本书的篇幅创建了一个可以实际工作的 Web 站点。但是，本书的每一章都是独立的，分别介绍了各个模块的开发过程，您可以把这些模块用于自己的 Web 站点中。书中还推荐了一个基本的站点构建框架，您可以利用该框架方便而快捷地创建一些模块，并把它们插入到 Web 站点中。

本书内容

本书各章的讲述遵循提出问题、设计方案和解决方案的顺序。首先提出需要解决的问题，然后说明实现的过程，最后在 Visual Studio .NET 中编写具体的软件代码。

大多数章节都涉及到 3 层系统的创建，包括数据层、业务层和表示层。我们还将介绍如何创建不同的模块，使它们能集成到站点中。

第 1 章概述了所要创建的 Web 站点。首先提出要解决的问题，并且讨论解决问题的过程和步骤。然后提出解决方案，涉及到后续各章中创建和集成模块的具体内容。

第 2 章介绍站点的构造基础。本章将建立编码标准，并设计文件夹和命名空间的结构。我们创建初始的数据库，但在这一阶段数据库中还没有数据。我们还创建了整个站点的错误处理代码、数据的基类和业务层对象。

第 3 章介绍把基础构造扩展到表示层。我们将为站点中的 ASP.NET 页面创建基类、自定义的错误提示页面、整个站点的导航、题头与脚注控件。

第 4 章创建文件管理模块，该模块用于下载和加载站点的源代码，并能够进行在线修改。我们还将介绍 Microsoft 的 Data Manager，它可以用来管理站点中的 SQL Server 数据库。

第 5 章介绍用户账号模块。本章介绍如何创建功能强大的基于角色的安全系统，并且把它集成到 ASP.NET 的内置验证功能中。

第 6 章介绍如何对 Web 站点的新闻内容进行定期修改。我们还将创建一个 Web 服务，把



新闻标题显示给其他站点和应用程序，并创建一个使用该 Web 服务的 Windows 新闻接收器。

第 7 章介绍站点的广告处理模块。本章将扩展 ASP.NET AdRotator 控件，提供需要的功能。还将介绍如何记录广告的点击率和用户印象，以及如何为广告商提供报告。

第 8 章介绍民意调查。本章将介绍如何管理用户的询问，记录用户的投票数，并且把它们整理成有用的报告。

第 9 章提供了创建电子邮件简讯的工具。本章将介绍如何创建纯文本和 HTML 格式的消息，以及如何管理消息列表和创建新消息。

第 10 章介绍论坛。本章将创建支持用户发送和阅读消息的全部功能，并且为管理员提供特殊的许可。本章将展示 DataList 和 DataGrid 控件的强大功能。我们还将介绍如何使用正则表达式来提供有限的 HTML 支持，使得站点不会遭受跨站点脚本的风险。

第 11 章介绍如何部署站点。我们将介绍如何利用 Visual Studio .NET 提供源代码免费的、可发布的软件版本，以及如何把站点部署到主服务中。

第 12 章展望未来。我们刚刚开始 ASP.NET Web 站点开发人员的生涯，Wrox 将会为读者提供不间断的支持，包括本书的 P2P（程序员对程序员）列表，您可以积极参与，和其他热心读者共同分享开发理念和经验。

本书读者对象

本书是为具有一定的 ASP.NET 基础，并且希望应用这些知识来开发 Web 站点的读者编写的。如果您已经阅读过了 Wrox 的《ASP.NET 入门经典——Visual Basic .NET 编程篇》、《ASP.NET 高级编程》和 VB.NET 方面的图书，阅读本书就会更加方便。

您应该熟悉如何使用 Visual Studio .NET 创建 ASP.NET 项目，并对 Visual Basic .NET 有一定的了解。

学习本书的软硬件需求

运行本书的代码示例需要以下条件：

- Windows 2000 或者 Windows XP。
- Visual Studio .NET 1.0。我们已经测试了 1.0 版本下的代码运行情况，大多数代码也可以在后续发布的版本中运行。这些代码都可以在 Visual Basic .NET 标准版下正常工作。
- SQL Server 2000，但本书使用的大多数技术也可以用于其他数据库系统，包括 Access。

为了让站点正常工作，您可能还需要 ASP.NET web 主机。本书最后将会提供一些选择 Web 主机的建议。

客户支持

我们一向重视您的反馈意见，希望知道您对本书的看法：批评、赞扬和建议都欢迎。您发

送反馈意见的电子邮件地址是 feedback@wrox.com。

如何下载本书的示例代码

访问 Wrox 站点 <http://www.wrox.com/>，只须从 Search 栏输入书名，或者使用书名列表。单击 Code 栏的 Download，或者在本书的详细信息页面中单击 Download Code。

站点中可供下载的文件使用 WinZip 格式压缩。读者在硬盘的文件夹中保存附件后，需要使用 WinZip 或者 PKUnzip 解压缩。解压缩文件后，代码通常解压缩到相应章的文件夹中。开始解压缩时，要确保 WinZip 或 PKUnzip 设置的文件夹名正确。

勘误表

尽管我们努力确保本书的文本和代码准确无误，但是仍旧会有挂一漏万之处。您若发现书中的错误，如拼写错误或者代码运行错误，请把勘误表发往电子邮件地址 support@wrox.com。

在 <http://www.wrox.com/> 站点可以找到勘误表，通过 Advanced Search 或者书名列表找到书名，单击在本书详细信息页面的图像下面的 Book Errata 链接。

电子邮件支持

如果您希望就书中的问题直接向掌握详情的专家询问，可以发电子邮件到 support@wrox.com，电子邮件中要有书名和 ISBN 书号的后 4 位数字。通常电子邮件中要包括如下内容：

- 书名和 ISBN 书号的后 4 位数字，主要问题所在的页码。
- 您的姓名、联系方法，以及主要问题。

我们不会向您寄送无用邮件，这样可以节省双方的时间。您的电子邮件一般会经过以下支持过程：

- 客户支持 您的电子邮件首先寄送给我们的客户支持人员，他们能够对常见的一些问题进行解答，并能就书中或者 Web 站点的常见问题立即作出解答。
- 编辑支持 较难的问题会送给负责本书的技术编辑，他们熟悉编程语言和软件产品，能够回答相关的细节性技术问题。
- 作者支持 最后，对于编辑无法解答的问题，将转发给作者。尽管我们不愿意干扰作者的写作工作，但仍旧会向他们转发一些具体的问题。全体 Wrox 作者都会对他们的著作提供技术支持。他们会向客户支持人员或者编辑发送电子邮件，使得所有读者都能受益。

Wrox 的技术支持工作只对直接与书中内容相关的问题负责，对于超出正常支持范围的问题，可以通过 <http://p2p.wrox.com/> 论坛上的讨论列表获得支持。

p2p.wrox.com

作者之间的讨论已加入到 P2P 邮件列表中。我们的独一无二的系统通过邮件列表、论坛和新闻组提供了程序员到程序员（programmer to programmer）之间的联系，这是一种一对一的电子邮件支持系统。您向 P2P 发送电子邮件查询时，会被许多 Wrox 作者和当时在邮件列表中的业界专家看到。您可以从 p2p.wrox.com 站点找到许多不同的邮件列表，它们不仅对阅读本书



有用，对您开发实际的应用程序也大有帮助。对于本书比较适合邮件列表是 `aspdotmet_website_programming`。

订阅邮件列表按照以下步骤：

- (1) 进入站点 <http://p2p.wrox.com/>。
- (2) 从左侧的菜单栏中选择合适的类别。
- (3) 单击希望加入的邮件列表。
- (4) 遵循相关指令进行订阅，并输入你的电子邮件地址和密码。
- (5) 确认希望接受电子邮件。
- (6) 使用订阅管理器加入更多的邮件列表，并设置电子邮件首选项。

本系统为什么能提供最佳支持

您可以加入邮件列表，并每周接收一次。如果您没有时间，或者不方便接收邮件列表，可以搜索我们的在线档案室。垃圾邮件已经被删除，您的电子邮件地址由独一无二的 Lyris 系统保护。任何加入或退出邮件列表订阅的请求，或者其他有关邮件列表的请求，都会发送到 listsupport@p2p.wrox.com。

目 录

第 1 章 建立 ASP.NET Web 站点	1
1.1 提出问题	1
1.2 设计方案	2
1.2.1 在不同地方工作	2
1.2.2 可维护、可扩展的站点	2
1.2.3 组建社团	3
1.2.4 吸引人的内容	3
1.2.5 广告	3
1.2.6 频繁的访问	4
1.3 解决方案	4
1.3.1 在不同地方工作	4
1.3.2 创建可维护、可扩展的站点	4
1.3.3 提供吸引人的内容	4
1.3.4 管理广告	5
1.3.5 激励社团	5
1.3.6 让用户重复访问	5
1.3.7 部署站点	5
1.4 小结	5
第 2 章 基础设计	7
2.1 提出问题	7
2.2 设计方案	10
2.2.1 命名和编码约定	10
2.2.2 编程语言	14
2.2.3 文件夹结构	15
2.2.4 设计数据库	16
2.2.5 建立数据服务层	17
2.2.6 建立业务服务层	18
2.2.7 错误处理机制和容错性	19
2.2.8 部署问题	19
2.2.9 用户界面设计	20



2.3	解决方案	20
2.3.1	DBObject 类	21
2.3.2	BizObject 类	26
2.3.3	AppException 类	26
2.4	小结	29
第 3 章	样式和导航基础	30
3.1	提出问题	30
3.2	设计方案	31
3.2.1	级联样式表	32
3.2.2	XSLT	34
3.2.3	页面基类	35
3.2.4	可重用的导航控件	35
3.2.5	题头和脚注	36
3.2.6	错误处理机制	37
3.3	解决方案	37
3.3.1	样式	38
3.3.2	PhilePage 类	40
3.3.3	导航器控件	41
3.3.4	SiteHeader 控件	46
3.3.5	SiteFooter 控件	48
3.3.6	错误处理	49
3.3.7	主页草图	51
3.4	小结	54
第 4 章	维护站点	55
4.1	提出问题	55
4.1.1	FTP 与在线文件管理	56
4.1.2	数据库管理	56
4.2	设计方案	57
4.2.1	实现设计方案	57
4.2.2	安全性设计	58
4.3	解决方案	58
4.3.1	处理文件和目录的类	58
4.3.2	题头和脚注控件	59
4.3.3	文件管理器的主页	60
4.3.4	使用 Windows 安全机制保护文件管理器	118
4.4	小结	121

第 5 章 用户和验证	122
5.1 提出问题	122
5.2 设计方案	123
5.2.1 窗体验证机制	124
5.2.2 规划数据库	124
5.2.3 数据层	126
5.2.4 业务层	128
5.2.5 模块配置	131
5.2.6 管理	133
5.3 解决方案	133
5.3.1 数据库	133
5.3.2 数据层	135
5.3.3 业务层	152
5.3.4 配置类	164
5.3.5 修改用户界面来支持验证	167
5.3.6 管理角色和账号	173
5.4 小结	181
第 6 章 新闻管理	182
6.1 提出问题	182
6.2 设计方案	183
6.2.1 要实现的功能	183
6.2.2 数据库设计	184
6.2.3 设计数据层	187
6.2.4 设计业务层	189
6.2.5 保存和检索设置	192
6.2.6 设计表示层	193
6.2.7 安全性需求	194
6.3 解决方案	195
6.3.1 处理数据库	195
6.3.2 实现数据访问程序集	202
6.3.3 配置程序集	209
6.3.4 实现业务层类	213
6.3.5 用户界面	218
6.3.6 向用户显示新闻	246
6.3.7 用户提交的新闻	250
6.3.8 要闻标题用户控件	251
6.3.9 保护模块	254



6.3.10 要闻标题 Web 服务	256
6.4 小结	263
第 7 章 广告	264
7.1 提出问题	264
7.1.1 印象	264
7.1.2 点击数	264
7.1.3 需求	265
7.2 设计方案	265
7.2.1 使用 AdRotator	266
7.2.2 数据库	267
7.2.3 数据服务层	268
7.2.4 业务层	269
7.2.5 表示层	269
7.2.6 配置	269
7.3 解决方案	269
7.3.1 数据库表	270
7.3.2 数据服务层	273
7.3.3 业务层	280
7.3.4 表示层	284
7.3.5 模块配置	292
7.4 小结	292
第 8 章 民意调查	294
8.1 提出问题	294
8.2 设计方案	296
8.2.1 需要实现的功能	296
8.2.2 设计数据库表	298
8.2.3 管理数据库的存储过程	300
8.2.4 设计数据层	301
8.2.5 存储和检索设置	303
8.2.6 设计业务层	304
8.2.7 设计用户界面服务	306
8.2.8 安全性需要	308
8.3 解决方案	309
8.3.1 处理数据库	309
8.3.2 实现数据访问程序集	315
8.3.3 配置程序集	315
8.3.4 业务程序集	318

8.3.5	管理用户界面	323
8.3.6	管理选项	334
8.3.7	在线修改设置	336
8.3.8	民意调查用户控件	339
8.3.9	ShowArchive.aspx 页面	354
8.3.10	安全模块	358
8.3.11	民意调查 Web 服务	358
8.4	小结	363
第 9 章	邮件列表	365
9.1	提出问题	365
9.2	设计方案	367
9.2.1	设计数据库表	367
9.2.2	管理数据库的存储过程	369
9.2.3	设计数据服务	371
9.2.4	存储和恢复设置	373
9.2.5	设计业务服务	374
9.2.6	设计用户界面服务	378
9.2.7	安全需要	379
9.3	解决方案	379
9.3.1	数据库的操作	379
9.3.2	实现数据和配置程序集	383
9.3.3	实现业务层	383
9.3.4	管理用户界面	394
9.3.5	在线修改设置	401
9.3.6	订阅邮件列表	406
9.3.7	保护模块	408
9.4	小结	409
第 10 章	论坛和在线社团	411
10.1	提出问题	411
10.2	设计方案	412
10.2.1	数据库	412
10.2.2	设计数据服务	418
10.2.3	配置	418
10.2.4	业务层	419
10.2.5	设计用户界面层	420
10.3	解决方案	421
10.3.1	创建数据库表	422



10.3.2	实现数据层	427
10.3.3	实现配置层	427
10.3.4	实现业务层	428
10.3.5	表示层	435
10.4	小结	470
第 11 章	部署站点	472
11.1	提出问题	472
11.2	设计方案	472
11.2.1	部署数据存储	472
11.2.2	为部署准备站点	473
11.2.3	手工(XCopy)部署	474
11.2.4	主机服务部署	475
11.2.5	自动部署(创建安装程序)	476
11.3	解决方案	477
11.4	小结	480
第 12 章	结束语	481
12.1	开始创建	481
12.2	加入我们的团体	481
12.3	请参考更多的书	481
12.3.1	Web 服务	481
12.3.2	安全	482
12.3.3	ADO.NET	482
12.3.4	服务器控件	482

第1章 建立ASP.NET Web 站点

本书将介绍如何建立基于内容的 ASP.NET Web 站点。这种 Web 站点由许多模块组成，它们协同工作来生成最终的站点。

本书将按照下面的标准顺序来建立每一个模块：

- 提出问题：需要完成什么工作？有哪些限制因素和其他因素需要考虑？
- 设计站点：确定解决问题时所需要的功能，获得一个解决方案的工作蓝图。
- 建立解决方案：编写代码和其他内容，实现设计方案。

本书主要解决编程问题。我们在提到设计时，一般指的是软件的开发，而不是着眼于图形或者用户界面的设计。

用户的 Web 站点不是解决与我们要求相同的所有问题，但是这里建立的模块和使用的编程技术对于用户的工作仍然是非常有益的。

本章将概括地介绍一下整个站点——需要完成的工作，以及这些工作的完成方式。

1.1 提出问题

我们要为 DVD 和图书爱好者建立一个 Web 站点。要想提出本站点的问题，需要考虑站点的目标和用户。在现实生活中，这一阶段是面向商务的，需要考虑诸如广告的观众统计、竞争对手和投资的可行性。这些问题需要严格的分析，不过这些还是留给职业经理们去解决吧。

本站点要尽量迎合 DVD 和图书的爱好者。该站点将提供有用的内容并试图建立自己的社团。访问者希望阅读有关的内容，提出他们的建议，但是每一次访问都比较短，因此不会像 Internet Movie Database (www.imdb.com)那样需要庞大的数据库。本站点由广告赞助，依赖于读者的重复(但是比较短)访问。

我们还要考虑会受到哪些限制，这是很实际的问题。本站点面临的一个主要限制就是它的开发小组，开发小组的成员们彼此不会碰面，因为他们遍布世界各地。这意味着站点的设计方案必须允许完成不同模块的站点开发人员彼此互不干扰，但是所有模块最终都要顺畅地工作。大多数情况下，开发人员彼此隔离不容易走极端，但是让每一个开发人员都能够独立地工作会非常高效。我们需要设计并建立一些方法来适应这一情况。

站点的开发永远不可能一蹴而就，总是要不断地修改。Web 站点成功的另一个关键是易于修改。我们需要找到合适的方法。

说明：

我们的站点名为 ThePhile.com，因为它是图书爱好者(藏书家)和 DVD 爱好者(DVD-philes)的站点。它的名字还暗含了 file(文件)，因为我们的站点将成为权威的信息源。



1.2 设计方案

前面列出了站点的需求，下面看看该站点能够提供什么功能。在提出问题部分主要涉及了以下几点内容：

- 支持开发人员在不同地方工作。
- 建立一个可维护、可扩展的站点。
- 建立社团。
- 提供有趣的内容。
- 通过广告获得收益。
- 鼓励频繁的访问。

下面依次讨论这些问题。

1.2.1 在不同地方工作

开发人员需要在交流较少的情况下建立站点的各个模块。开发人员位于不同的国家，因此面对面地开会是不可能的。电话交谈太费钱，时差问题也会带来不便。

系统的设计方案需要支持开发人员独立地完成他所承担的站点模块，而又不会妨碍其他开发人员的工作。

解决这一问题的一种好办法是以模块系列的方式进行开发，每一个模块都是相对独立的。当然，也会有共享的组件，但是对共享组件的修改很少，而且修改的方式是可控制的。本书将以模块化方式工作。我们还会频繁使用控件一词。控件在这里指的是页面的组件，它们可以独立地开发，并便于在需要时拖放到页面中。对于站点中实际页面的修改要尽量避免。

1.2.2 可维护、可扩展的站点

大多数 Web 站点都需要不断地添加新功能。因此站点在设计之初就需要使修改工作易于进行。

以模块化方式工作并使用控件可以在某种程度上简化修改工作。实际上，使用控件意味着非程序员也可以比较容易地编辑站点的页面——他们看到的只是 HTML 代码。控件就像是另一种 HTML 标记一样。

以模块化方式工作意味着，可以随时向站点中添加新的模块，而不会妨碍整个站点。每一个模块都是彼此独立的，因此可以方便地添加新的模块，修改原有的模块。

每一个独立的模块都需要能够方便地修改。完成这一目标的一种好办法就是在层(layer 或者 tier)上工作。大多数模块都将使用 3 层的设计模式。它们是数据层(data layer)、业务层(business layer)和表示层(presentation layer)。数据从数据层传递到业务层，再从业务层传递到表示层，然后向回传递。每一个层都有自己的任务。数据层的下面是数据源，数据层的任务就是访问数据源。

数据层从数据库中获得比较原始的数据(例如“-10”)。业务层把数据转换成符合业务规则的有意义的信息(例如“零下 10C”)。表示层把信息转换成对用户有意义的内容(例如“哇！要

冻僵了!”)。

这种方式很有用,因为每一层都可以独立地修改。我们可以修改业务层,如果从数据层接收相同的数据,并把这些数据传递到表示层,就不必担心出现歧义。我们也可以修改表示层,改变站点的外观,而不必改动下面的业务层逻辑。

这意味着可以向不同的用户提供不同的站点版本。而这只需要调用相同业务层对象的不同表示层。例如,为不同国家的用户提供不同的问候语和鼓励语。

我们需要一些方法来对活动的站点进行修改。这可以通过 FTP 上载完成,但是大多数时候通过 Web 界面会更好。

我们还需要管理其他模块的工具,例如禁止有问题的用户访问站点、添加新的文章等。这些工作也是一个可维护的站点必须要做的。

1.2.3 组建社团

站点通常能够从用户的参与中获益。由于本站点不希望用户花费大量的时间在线阅读,因此我们的社团功能也不能占用用户的大量时间。

组建社团的方式有两种:通过民意调查或者在线论坛。民意调查可以让用户有机会通过单击鼠标发表意见,这样做占用用户的时间很少,但是却可以让站点充满生机。

论坛可以让用户和其他用户讨论某项议题。消息保留在系统中,回复传递给用户。您在发出消息后,可以过一段时间再看看有没有回复。对于我们的目标而言,这样做比建立聊天室要好,因为聊天室要求读者在整个聊天的过程中都要保留对站点的连接。

社团可以给站点赋予生命。过一段时间以后,社团中就会涌现出各式各样的成员。许多站点都完全依赖于社团,并且会变得极其流行,例如 www.plastic.com。

为了组建社团,需要标识出不同的用户,并且为他们分配唯一的登录号。因此,系统需要某种形式的用户账号系统。

1.2.4 吸引人的内容

本站点的用户最关心的是与电影和图书相关的新闻和评论。这些内容的时效性很强,可能会集中在一部小说出版或者突然流行的很短一段时间里。我们的站点需要以这种方式管理新闻的工具。

提供吸引人的内容还有另外一种方式,就是让用户自己提供。这是社团模块要完成的工作之一。组建社团的目标之一就是让用户自己提供内容。

1.2.5 广告

广告为我们提供收入(有时也用来和其他站点交换网页广告横幅)。我们需要显示广告,并且记录每一条广告显示和点击率。

我们还需要收集站点用户的喜好信息,以便定位广告的内容。民意调查和在线论坛能够提供要做广告的产品有用信息。

大的站点都是基于用户统计数据和其他各种有关用户的信息来定位用户的(例如, Yahoo! 和 Amazon.com Web 站点的广告定位和产品推荐就是依靠用户统计数据和用户的购买嗜好)。本



站点的用户定位比较集中，而且用户群不是很大，因此不需要做这件工作。

1.2.6 频繁地访问

好的站点应该能让用户再度光临。如果站点的内容引人入胜，又有许多在线论坛，用户就会频繁地访问站点。

不时地提醒用户也不错。我们希望用户即使没有访问站点时也能够关注我们的站点。方法之一就是向用户发送电子邮件简讯，给用户提供有用的信息，并恰当地提醒他们访问站点。

我们还将创建一个 Windows 应用程序作为新闻更换器，自动地更新新闻标题。用户可以单击标题，阅读站点中完整的新闻内容。

1.3 解决方案

前面介绍了站点的主要任务，并勾画出了这些任务的实现方式。下面说明如何建立解决方案。这实际上是本书的主要内容。下面介绍一下本书各章与前面所述问题与设计方案的联系。

1.3.1 在不同地方工作

在接下来的两章中将提供一个开发的框架，涉及到编码标准、模块的文件夹组织框架和 Visual Studio .NET 项目。

我们将确定每一个模块所使用的命名空间，以及让小组开发顺利进行的其他事项。我们还将开发一些跨站点使用的初始的用户界面功能，使站点的外观统一。界面部分包括题头、脚注、导航控件和样式表等。

1.3.2 创建可维护、可扩展的站点

第 2 章和第 3 章还将考虑站点的可维护性。我们将开发基类，为每一个新模块提供坚实的创建基础。

第 4 章将开发一个基于 Web 的文件管理器，用来下载和上传文件、创建新文件、移动文件、修改文件属性，甚至可以利用一个内置的、基于 Web 的文本编辑器在线编辑文件。如果用户希望提供文件上传功能，提供下载的源代码，或者提供在线编辑工具，这个文件管理器会非常有用。

我们开发的大多数模块都具有管理功能。为了有效地进行管理，需要标识管理员。第 5 章将开发一个用户账号系统，用来收集用户信息，并为不同的用户提供不同的权限。最终的站点将支持完全基于角色的安全机制，登录信息保存在 SQL Server 数据库中。

1.3.3 提供吸引人的内容

第 6 章将创建一个新闻管理系统，让管理员添加和编辑新闻文章，接收和批准用户推荐的新闻文章，对新文章进行分类。当然，该系统还要允许用户阅读新闻。我们将创建一个控件，以便在任何需要的页面中方便地显示新闻标题。

新闻系统要有充分的灵活性来容纳评论，这是我们站点的核心功能。