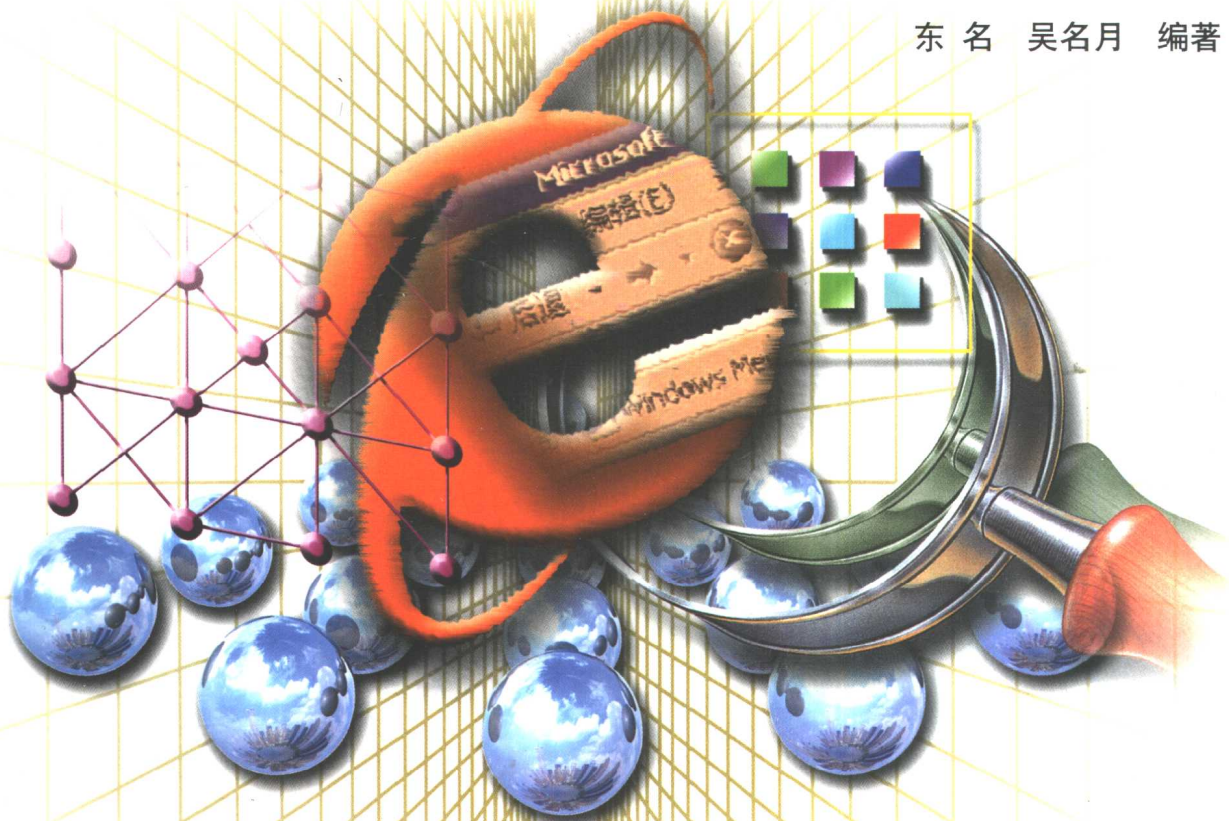


ASP.NET

动态网页设计高手

东 名 吴名月 编著



清华大学出版社
<http://www.tup.tsinghua.edu.cn>



ASP.NET

动态网页设计高手

东名 吴名月 编著

JS88-10

清华大学出版社

(京)新登字 158 号

内 容 简 介

ASP.NET 是 ASP 的升级版,它采用了真正的程序语言。ASP.NET 支持的程序语言有 JavaScript、Visual Basic.NET 和 C#, 功能更强大。

本书着重介绍 ASP.NET 的新网页技术,其中包括过去 ASP 所没有的 HTML 控件、Web 控件以及许多网页设计高级技巧。本书也介绍了 ADO.NET 如何与数据库连接、数据集的相关操作以及将 ADO.NET 与 DataGrid、DataList、Repeater 这 3 种功能强大的数据控件相结合,让程序员更有效率地开发各类数据库应用程序。

最后,本书以“网络书店”为主题,结合书中所介绍的各种新功能,介绍了一个实例,并配以详细的说明,学完后使读者很快成为 ASP.NET 高手。

本书适合于已具备 ASP 程序设计初步知识的程序开发人员及管理、维护人员使用。

本书繁体字版名为《ASP.NET 快速升级秘笈》,由文魁资讯股份有限公司出版,版权属东名、吴名月所有。本书简体字中文版由文魁资讯股份有限公司授权清华大学出版社独家出版。未经本书原版出版者和本书出版者书面许可,任何单位和个人均不得以任何形式或任何手段复制或传播本书的部分或全部内容。

北京市版权局著作权合同登记号:图字 01-2001-3498 号

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。

书 名: ASP.NET 动态网页设计高手

作 者: 东名 吴名月

责任编辑: 高志凤

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.tsinghua.edu.cn>

印 刷 者: 北京牛山世兴印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×960 1/16 **印张:** 25.75 **字数:** 531 千字

版 次: 2001 年 10 月第 1 版 2001 年 10 月第 1 次印刷

书 号: ISBN 7-302-04869-X/TP·2891

印 数: 0001~5000

定 价: 38.00 元

前 言

ASP.NET 是 ASP 的升级版，它支持的程序语言有 JavaScript、Visual Basic.NET(简称 VB7，是 Visual Basic 6 在 .NET 平台上的改良版)和 C#(结合 Java、C++、Visual Basic 特性的新程序语言)，其功能更强大。

本书结构合理，精心编排为：基础篇、网页技术篇、数据存取篇、对象开发篇与应用篇。

基础篇：主要介绍 ASP.NET 的基本概念、安装 .NET Framework SDK；比较 ASP 与 ASP.NET，并且提供升级版本中应注意的事项；针对过去用 VBScript 编写 ASP 程序的读者，简略地比较了两种程序语言的差别，并提供一些升级时的小技巧等。

网页技术篇：这一篇着重介绍 ASP.NET 的新网页技术，其中包括旧版 ASP 所没有的 HTML 控件、Web 控件以及许多网页设计高级技巧。

数据存取篇：本篇也是本书着墨最多的部分，介绍了 ADO.NET 如何与数据库连接；数据集的相关操作，包括表的建立、数据记录的添加、删除、修改、查询等议题；如何将 ADO.NET 的应用与 DataGrid、DataList、Repeater 这 3 种功能强大的数据控件结合，让程序开发人员高效地开发各类数据库应用程序。

对象开发篇：介绍用户如何自行开发对象，以扩充功能和 Web 服务。

应用篇：以“网络书店”为例，结合本书中所介绍的各种新功能，并配以详细的说明，对 ASP.NET 进行更进一步的讲解。

目 录

第 I 篇 基础篇

第 1 章 ASP.NET 概述	3
1.1 ASP 与 ASP.NET	4
1.1.1 程序代码的结构	4
1.1.2 控件模型	5
1.1.3 语言支持	5
1.2 ASP.NET 与 .NET 平台的结合	6
1.2.1 ASP.NET 与 .NET Framework	6
1.2.2 ASP 与 ASP.NET 的兼容性	7
1.2.3 ASP.NET 网页服务结构	7
1.3 ASP.NET 网页开发控件	9
1.3.1 HTML 控件	9
1.3.2 Web 控件	11
1.4 ASP.NET 数据库开发	12
1.5 安装 .NET Framework	15
第 2 章 ASP 与 ASP.NET 的区别	19
2.1 程序结构	20
2.1.1 程序区段	21
2.1.2 使用的程序语言	22
2.1.3 事件目录	24
2.2 安全性	25
2.3 输出高速缓存	26
2.3.1 ASP.NET 输出高速缓存功能设置	27
2.3.2 有效期限类型	29
2.4 数据存取	29

2.4.1	处理数据记录.....	30
2.4.2	数据控件	34
2.5	ASP.NET 的性能	35
2.5.1	早绑定与迟绑定.....	35
2.5.2	调用 Managed Code 与 Unmanaged Code 代码	35
第 3 章	Visual Basic 6 与 Visual Basic.NET 的区别	37
3.1	新的开发平台——.NET Framework	38
3.2	Visual Basic.NET 程序编译方式	38
3.3	面向对象程序设计	40
3.4	程序语法的改进	40
 第 II 篇 网页技术篇 		
第 4 章	HTML 控件	49
4.1	准备工作	50
4.1.1	设置网页属性.....	50
4.1.2	ASP.NET 的新技术——服务控件.....	51
4.2	认识 HTML 控件.....	51
4.2.1	数据处理	51
4.2.2	事件对象	52
4.3	使用 HTML 控件.....	55
4.3.1	HTML 控件的结构	55
4.3.2	常用的 HTML 控件	56
4.4	表单应用	74
第 5 章	Web 控件.....	79
5.1	认识 Web 控件.....	80
5.2	使用 Web 控件.....	81
5.3	HTML 控件与 Web 控件	108
第 6 章	网页设计高级技巧	109
6.1	高级的控件	110
6.2	验证控件	123

6.3 网页应用程序环境	138
6.3.1 ASP.NET 的组合设置	138
6.3.2 使用全局声明	146
6.3.3 使用 ASP.NET 的高速缓存提高运行性能	148
6.3.4 ASP.NET 网页的跟踪与调试技巧	151

第III篇 数据存取篇

第7章 ADO.NET 数据库连接	159
7.1 ADO.NET 概述	160
7.1.1 ADO.NET 的结构与运作方式	160
7.1.2 ADO.NET 的主角——DataSet 对象	161
7.1.3 ADO.NET 数据库连接	162
7.1.4 ADO.NET 名称空间	162
7.2 从 ADO 到 ADO.NET	164
7.2.1 ADO 传统运作模式	164
7.2.2 ADO.NET 的优势	165
7.3 ADO.NET 数据连接技巧	166
7.4 使用 ADOConnection 对象	170
7.5 使用 ADOCommand 对象	171
7.6 使用 ADODatasetCommand 对象	175
7.7 使用 ADODataReader 对象	176
第8章 ADO.NET 数据管理	181
8.1 数据集与数据管理	182
8.2 使用 DataTable 对象	184
8.3 使用 DataColumn 对象	201
8.4 使用 DataRow 对象	207
8.5 使用 DataRelation 对象	209
8.6 使用 Constraints 对象	214
8.7 使用 DataView 对象	217
第9章 数据控件	227
9.1 数据控件的特性	228

9.2 DataGrid 控件	229
9.2.1 自定义 DataGrid 外观	231
9.2.2 设置分页功能	234
9.2.3 设置排序功能	242
9.2.4 设置字段类型	247
9.2.5 编辑 DataGrid 数据	258
9.3 DataList 控件	265
9.3.1 DataList 基本外观设置	266
9.3.2 7 种 DataList 模板	269
9.3.3 DataList 高级外观设置	273
9.3.4 DataList 数据选取功能	279
9.3.5 DataList 数据编辑功能	281
9.4 Repeater 控件	287

第IV篇 对象开发篇

第 10 章 对象开发	293
10.1 开发用户控件	294
10.1.1 建立 Pagelets 控件	294
10.1.2 建立 Class 控件	303
10.2 建立.NET 的链接库	306
第 11 章 Web 服务	315
11.1 Web 服务概述	316
11.2 建立 Web 服务	318
11.3 使用 Web 服务	320
11.4 Web 服务所支持的数据类型	325

第V篇 应用篇

第 12 章 网络书店开发实例	333
12.1 网络书店运作流程	334
12.2 数据库设计	340
12.3 系统结构	342

12.4	自定义类	342
12.5	书籍分类浏览	346
12.6	书籍详细介绍	349
12.7	购物车	351
12.8	收银台	356
12.9	订单	365
12.9.1	确认订单	365
12.9.2	保存订单	368
附录 A	网络书店源程序代码	373
附录 B	Regular Expressions 的语法	395

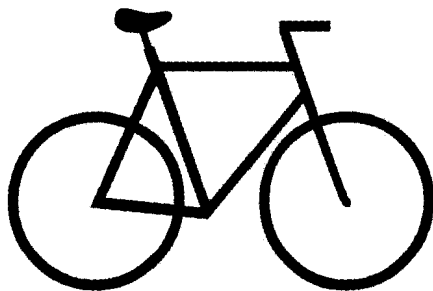


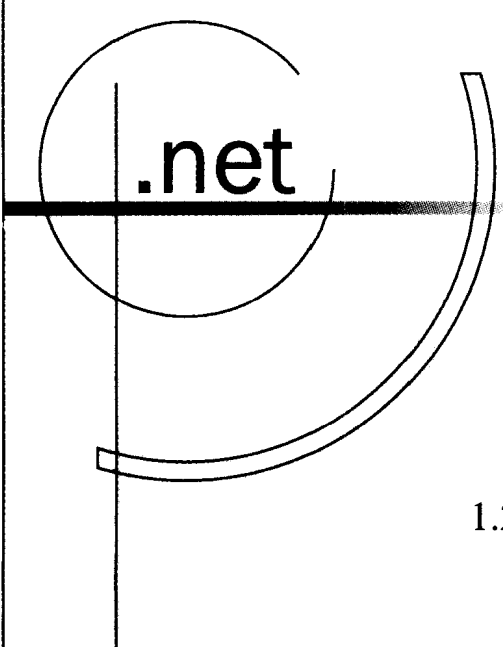
第 I 篇 基础篇 ○○○○○

第 1 章 ASP.NET 概述

第 2 章 ASP 与 ASP.NET 的区别

第 3 章 Visual Basic 6 与 Visual Basic.NET 的区别



A stylized logo for .NET. It features a circle with a horizontal line passing through its center. The text ".net" is positioned to the right of the circle, with the dot of the period overlapping the circle's edge. A vertical line extends downwards from the center of the circle, and a curved line arches over the circle from the left side.

.net

第 1 章

ASP.NET 概述

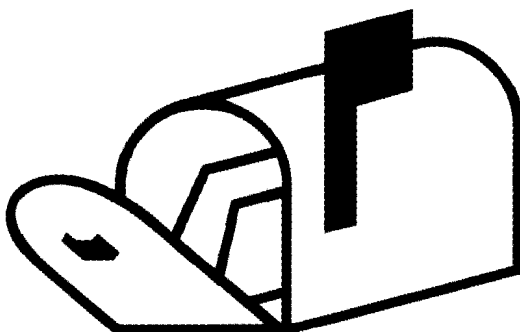
1.1 ASP 与 ASP.NET

1.2 ASP.NET 与 .NET 平台的结合

1.3 ASP.NET 网页开发控件

1.4 ASP.NET 数据库开发

1.5 安装 .NET Framework



1.1 ASP 与 ASP.NET

ASP.NET 是 ASP 升级的版本。ASP 虽然易学易用,但它简单的语法与较松散的结构,使得 ASP 一直有一些潜在的问题,下面将对这些问题进行说明。

1.1.1 程序代码的结构

一个程序如果没有合理的结构与设计风格会使程序显得零乱,而且不易懂,尤其是当进行大型项目时。相信大家都有这样的经验:虽然 ASP 很好写,但写完之后连自己都有点看不懂,何况是其他人要接手维护更是难事。下面是一些简单的 ASP 程序片断:

```
<HTML>
<Body Background=" Back.gif ">
.....
<% Sub TestProcedure() %>
    <H3> 显示 HTML 文本 </H3>
<% End Sub%>
.....
</Body>
</HTML>
```

ASP 网页总是将程序代码与 HTML 代码混杂在一起,这也表示 ASP 网页会将网页设计的部分(如美工、样式、图片、文字)与程序逻辑混合在一起,不但不易懂,而且也让开发人员难以将两样东西分开来处理。

现在 ASP.NET 已经大幅度改善了这种缺点,以上的写法在 ASP.NET 中将不被接受,而必须被改写为:

```
<script language=vb runat=Server>
    Sub TestProcedure()
        Response.Write(" <H3> 显示 HTML 文本 </H3> ")
    End Sub
</script>
<HTML>
.....
</HTML>
```

1.1.2 控件模型

ASP 没有面向对象的概念, 它顶多算是功能函数, 因此几乎每一个常用的小程序(例如数据分页功能)都必须一一去编写, 而无法直接引用对象的方式来达成。例如在 ASP 中为了要达到数据分页的功能, 可能要编写上百行的程序来处理, 而现在 ASP.NET 则可直接引用 DataGrid 服务器控件中的 Paging 属性来自动产生(如图 1.1)。

Product_ID	Product_Name	Product_Description
P001	IBM 笔记本	14' 屏幕, 500MHZ Intel CPU, 64 MB 内存
P002	IBM 台式机	17' 屏幕, 700MHZ Intel CPU, 128 MB 内存, 10G 硬盘空间
P003	Nokia 移动电话	Nokia 8110, 中文输入

运用 DataGrid 的 Paging 功能自动产生分页链接

上一页 下一页

图1.1 ASP.NET的Paging属性

相对于 ASP 而言, ASP.NET 的确让程序代码精简许多, 并减少重复编写相同功能的情形发生。除此之外, 也让 ASP 能够用真正的面向对象的方式编写, 如继承、建立对象等大部分与面向对象相关的操作, ASP.NET 都已经可以支持。

1.1.3 语言支持

大家都知道 ASP 所支持的语言只有 VBScript 与 JavaScript 两种, 但这两种语言都只能算是脚本语言(Script Language) 而不是真正的程序语言(Programming Language)。

这让 ASP 的功能受到很大的限制, 脚本语言易学易懂却功能有限, 这让大型项目的开发人员伤透脑筋, 因为语言功能的缺乏, 而让他们必须花费许多时间自行编写程序来弥补它的不足。

现在 ASP.NET 已经决定采用真正的程序语言作为开发的语言, 它支持功能强大的 VisualBasic.NET、C#, 只有 JavaScript 可以继续延用, 而 VBScript 已经被淘汰出局。如此一来, 任何人只要熟悉其中任一种语言, 就能胜任开发的工作, 并且功能也较过去强多了。

```
<Script Language="C#" runat="Server">    ←采用 C# 语法
<Script Language="Visual Basic" runat="Server">    ←采用 Visual Basic.NET
语言
```

表 1.1 ASP.NET 支持的程序语言

ASP.NET 支持的程序语言	说 明
JavaScript	惟一不变的语言
Visual Basic.NET	又称为 Visual Basic.NET, 为 Visual Basic 6 在.NET 平台上的改良版
C#	结合 Java、C++、VB 特性的新程序语言

1.2 ASP.NET 与.NET 平台的结合

1.2.1 ASP.NET 与.NET Framework

如果打开.NET Framework SDK 的说明文件, 用户会惊讶于.NET Framework 所提供的上百种好用的对象类别, 如图 1.2。

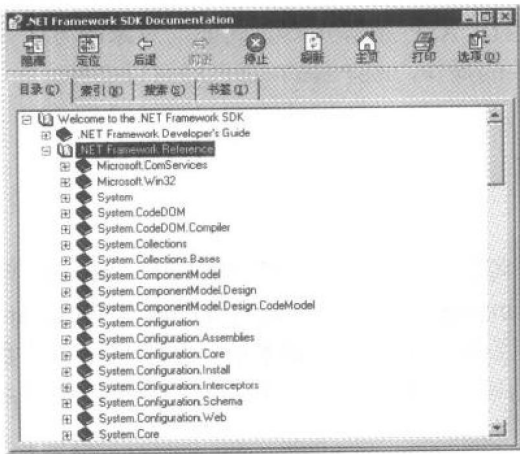


图1.2 .NET Framework的对象类别

.NET Framework 涵盖了大部分常用的功能, 这些只能用在一般的窗口程序中, 但只要通过适当的引用, 这些对象类别大部分都可以在 ASP.NET 网页中应用, 例如用户想开发数据库程序, 则可以引用 System.Data 这个名称空间:

```
<%@Import Namespace= " System.Data " %>
```

接着所有里面所涵盖的类别都可以使用, 对于程序员而言的确是轻松许多。

1.2.2 ASP 与 ASP.NET 的兼容性

很多人会问：“过去所开发的 ASP 网页是否还适用在新平台上”？其实微软也考虑到这个问题，为了让过去大家的努力不会白费，ASP 是可以与 ASP.NET 共存在 IIS 服务器中的。

将 .NET Framework 安装在服务器时，原来的 ASP 并不会因此受到影响，因为 ASP.NET runtime 会选择扩展名为 .aspx 的文件来处理(.aspx 即为 ASP.NET 网页程序的识别)，而 .asp 文件则会继续由原来的 ASP engine 来处理，如图 1.3 所示。

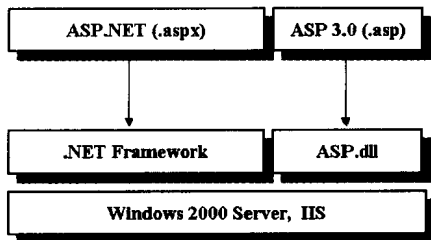


图1.3 ASP.NET与ASP 3.0的处理

从图 1.3 我们可以看到，原来的 ASP.dll 仍然存在，因此 ASP 仍可继续执行。但若想要享受新平台所带来的优势，就非得了解 ASP.NET 的运行，并将源程序稍做改写(有时几乎是全部重写，因为 ASP.NET 的确改进许多)，才可适用于新平台。本书在第 2 章中有许多相关技巧的介绍。

另外，ASP.NET 也采用真正的程序语言，以 Visual Basic.NET 取代 VBScript，相关的升级技术请参考本书的第 3 章。

1.2.3 ASP.NET 网页服务结构

ASP.NET 与 ASP 在系统内部的运行是有差异的。ASP 通过一个 ASP.DLL 动态连接文件来支持 ASP 的运行，其实相当简单；而 ASP.NET 则是与 .NET Framework 完全的整合。除了底层的 .NET Framework 所提供的各种对象都可以使用外，也另外提供 ASP.NET runtime 来处理应用程序相关的处理程序，包括 Session 对象、Application 对象、Output Cache 等功能。因为有了如此强大的支持，所以 ASP.NET 的功能增强许多，也大幅简化客户端的应用程序，读者可以参考图 1.4。

当然.NET Framework 并非专为 ASP.NET 所设计, 其他任何在.NET 平台上所开发的应用程序也都可以使用它。

除此之外, ASP.NET 让人印象深刻的是它会将程序代码编译成为中间语言(Intermediate Language, IL)来执行, 不论是用 Visual Basic.NET 或 C#, 当第一次被客户端请求(Request)时, 它会被编译成为 IL, 它是介于程序语言与机器码之间的一种语言, 只有在真正的执行时才会被转换为机器码来执行, 如图 1.5。

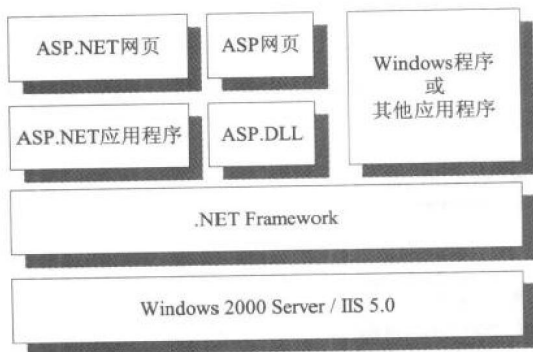


图1.4 ASP.NET与其他相关软件之间的关系

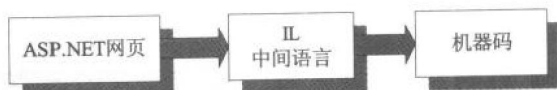


图1.5 ASP.NET的编译语言

IL 中间语言的特性是它与程序语言无关, 也就是说, 不论是用 Visual Basic.NET 或 C# 所编写的程序代码, 所产生出来的 IL 都是一样的。

为了不让每一次的网页请求都必须重新编译一次, 只有在第 1 次的请求进行编译至 IL 的操作, 并存储在高速缓存中; 以后都将直接执行 IL, 如此一来便可提升系统效率。只有当程序代码被修改后, 才会重新编译并更新高速缓存, 如图 1.6。

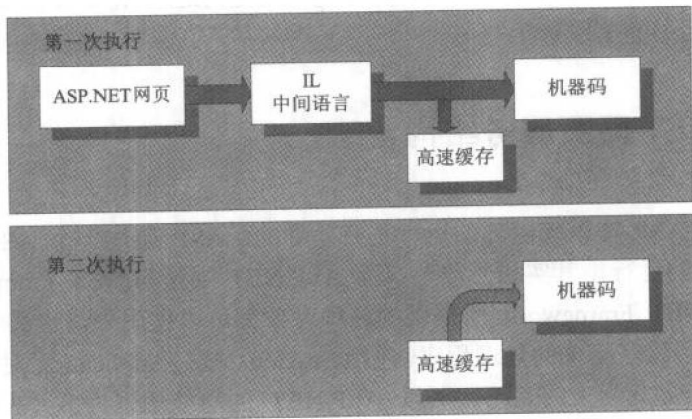


图1.6 ASP.NET的编译流程