数据库应用开发技术丛书

ASP.NET

数据库高级教程（C#篇）

李应伟
姚素霞　编著
景　丽

清华大学出版社

数据库应用开发技术丛书

ASP.NET 数据库高级教程(C#篇)

李应伟 姚素霞 景丽 编著

清华大学出版社

北 京

前　言

Visual Studio.NET 是 Microsoft 公司推出的可视化开发工具，ASP.NET 作为 Visual Studio.NET 的组成部分之一，已经成为 Internet 中 Web 应用程序的新一代开发工具，并逐渐被广大程序员普遍采用。

在微软的.NET 战略中，ASP.NET 是非常重要的一环，它相对于以前的 ASP 有了相当大的改进。相对于其他的 Web 应用开发模型来讲，ASP.NET 具有更大的优势，其主要特点包括：

- ASP.NET 与其前版 ASP 不同，它是在服务器上运行的编译好的公共语言运行时代码，可以更好地提高程序运行性能。
- ASP.NET 与语言无关，它可以采用 C#、VB.NET 以及 JScript 等支持.NET 框架的语言来进行开发，开发者可以根据自身的情况进行选择。
- ASP.NET 支持开发 Web 服务，它对 XML 技术提供了更好的支持，使得 Web 应用程序的开发更具有可扩展性和跨平台性。
- ASP.NET 提供了很多功能强大的服务器端控件，使得程序的开发更趋于简单化。同时，它还提供了 HTML 设计代码和后台编程代码分离的技术，并在后台代码开发中提供了强大的智能化支持，使得开发工作更具有条理性。

C#是一门崭新的语言，它具有开发效率高，应用范围广等特点，成为当前程序开发领域的一大热门。本书在开发 ASP.NET 应用程序的后台代码时就选用了 C#语言，相信其优秀的开发性能一定能给读者带来耳目一新的感觉。

本书主要介绍 ASP.NET 的高级技术，尽可能帮助开发人员解决实际开发项目中遇到的问题。它主要适用于 Web 应用开发初学者及广大网络设计和开发人员阅读，对高级开发人员也有一定的参考价值。

全书共分 20 章，第 1~11 章主要对 ASP.NET 高级技术的各个方面分别进行说明和解析，并附以实例指导；第 12~20 章则通过一些详实的开发程序来帮助读者深入理解 ASP.NET，以及学习如何利用 ASP.NET 来构建功能强大的 Web 应用程序。

第 1 章介绍 ASP.NET 中 Global.asax 文件。

第 2 章介绍应用程序配置，主要对 web.config 文件的使用进行了详细的说明。

第 3~5 章详细讲解 ASP.NET 中的数据访问技术，使得开发数据驱动的网站不再那么神秘。

第 6 章介绍了 XML 在 ASP.NET 中的使用，反映了 ASP.NET 对 XML 技术的支持。

第 7 章主要介绍 Web 服务，对 ASP.NET 中如何开发和使用 Web 服务进行了详细的说明。

第 8 章介绍 Web 窗体控件，自定义 Web 窗体控件等内容，说明控件在 Web 程序开发中的作用。

第 9 章介绍组件服务。

第 10 章和第 11 章分别涉及 ASP.NET 中的安全性与性能问题，可以用于开发更为高效、安全的 Web 应用。

第 12~20 章则通过几个大型的实例来分别介绍网站广告、图片处理、网络硬盘、邮件发送、统计图表、即时消息、BBS 系统的 Web 应用解决方案，对前面介绍的内容进行了一些综合的应用。

参加本书编写工作的人员有郭斌、李应伟、姚素霞、李翔、韦敏宗、田龙、钟遥、宋明颛和王微等。其中郭斌完成第 1、2、3、7、9、16 章，李翔完成第 5、11、12、18、19 章，韦敏宗完成第 6、10、14、17 章，田龙完成第 4、15 章，钟遥完成第 8、20 章，宋明颛完成第 13 章，王微完成了附录部分的写作工作。在写作的过程中大家都付出了相当多的努力，发扬了锐意进取的团队精神。

此外，蓝荣香、王昊亮、喻波、马天一、魏勇、郝荣福、李光龙、孙明、李大宇、武思宇、牟博超、付鹏程、高翔、朱丽云、崔凌、张巧玲、李辉、李欣、柏宇、郭强、金春范、程梅、黄霆、钟华、高海峰、王建胜、张浩、刘湘和邵蕴秋等同志在整理材料方面给予了作者很大的帮助，在此，对他们表示衷心的感谢。

但由于作者水平和经验有限，书中难免有不足之处，希望通过和广大的读者进行交流来解决，以使得本书在再版时更为完美。

作　者

2004 年 3 月

目　　录

第1章　ASP.NET应用程序设置

ASP.NET 应用程序被定义为可以从 Web 应用程序服务器上虚拟目录及其子目录中调用的所有文件、页、处理程序、模块和可执行代码的总和。在每一个 ASP.NET 应用程序里都包含一个名为 Global.asax 的文件。它主要负责一些高级别的应用程序事件，例如应用程序的开始和结束、会话状态的开始和结束等。

了解 Global.asax 文件的运行机制，将会对理解 ASP.NET 应用程序开发机理有很大的作用。本章将对 ASP.NET 应用程序以及 Global.asax 文件的配置进行介绍，希望这一章内容的讲述能为读者后面内容的学习奠定一个良好的基础。

1.1　Global.asax 概述

为了对 Global.asax 进行介绍，就必须首先安装 Visual Studio.NET 环境。本书的程序都是在 Visual Studio.NET 2003 中进行开发的，读者要先在自己的机子上搭建好开发环境。当然，也可以安装最新的 Visual Studio.NET 2003。为了创建一个 ASP.NET 应用程序，需要先安装并配置 IIS 服务器。一般来说，如果所安装的操作系统是 Server 版的话，则已经安装了 IIS 服务器；否则要先用安装盘来添加“IIS 服务组件”。安装好后，选择“开始”|“程序”|“管理工具”|“IIS 服务管理器”命令，就会弹出 IIS 配置对话框，并在这里完成虚拟目录的设置。另外还需要对相关文件夹进行设置，把其属性设置为“Web 共享”。这样就完成一个 ASP.NET 应用环境的设置。如果要添加一个新的 ASP.NET 应用，可以直接打开 Visual Studio.NET，添加新项目并设定项目名来实现，如图 1-1 所示。

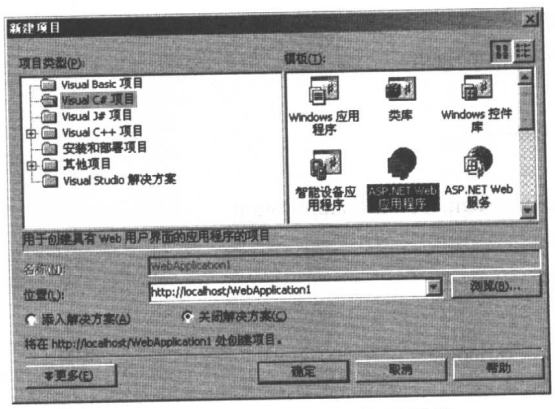

图 1-1　新建 ASP.NET 应用程序

在说明了如何创建 ASP.NET 应用程序后，就要来介绍 Global.asax 文件了。了解 ASP 的程序员都知道，在 ASP 中有一个 Global.asa 文件。该文件主要通过应用程序事件，如 Application_Start，Application_End，Session_Start 和 Session_End 等来处理应用程序逻辑。而在 ASP.NET 中则提供了 Global.asax。从概念上来说，这两者是非常类似的。当位于应用程序命名空间内的任何资源或者 URL 被首次访问时，ASP.NET 系统将自动解析 Glabal.asax 文件并把它编译为动态的.NET 框架类——此类派生自 HttpApplication 基类并加以扩充。在创建 HttpApplication 派生类实例的同时，还将引发 Application_Start 事件。随后，HttpApplication 实例将处理页面的一个个请求或者响应，同时触发 Application_BeginRequest 或者 Application_EndRequest 事件，直到最后一个实例退出时才引发 Application_End 事件。

Glabal.asax 文件是经过设置的。任何关于该文件的直接 URL 请求都将被拒绝，从而可以保证外部的用户无法下载及查看其内容。当改变 Glabal.asax 文件的时候，.NET Framework 框架可以及时地检测出这种变化。此时，它会完成应用程序的所有当前请求，将 Application_OnEnd 事件传送给每一位聆听者，重新启动应用程序。当浏览器发出的下一个请求到达时，ASP.NET 应用程序将重新解析 Glabal.asax 文件，并激发 Application_OnStart 事件。

Global.asax 文件是一个可选的文件，它位于 ASP.NET 应用程序根目录下。如果不对它进行定义的话，则系统默认为用户未定义任何应用程序或会话事件处理程序。一般来说，如果用户不需要编写任何应用程序或者会话事件处理程序的话，可以选择不要 Global.asax 文件。

下面通过一个程序实例来说明 Global.asax 文件的应用。首先，新建一个 ASP.NET Web 应用项目，其项目名称为 WebGlobal。其中，Global.asax 文件的代码设置如下：

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Web;
using System.Web.SessionState;

namespace WebGlobal
{
 public class Global : System.Web.HttpApplication
 {
      public Global()
      {
           InitializeComponent();
      }
      protected void Application_Start(Object sender, EventArgs e)
      {
```

```
    }
    protected void Session_Start(Object sender, EventArgs e)
    {
        Response.Write("Session Started<br>");
    }
    protected void Application_BeginRequest(Object sender, EventArgs e)
    {
        Response.Write("<h2>Global.asax Sample</h2>");
        Response.Write("Begin Reques...<br>");
    }
    protected void Application_EndRequest(Object sender, EventArgs e)
    {
        Response.Write("End Request...<br>");
    }
    protected void Application_AuthenticateRequest(Object sender, EventArgs e)
    {
    }
    protected void Application_Error(Object sender, EventArgs e)
    {
    }
    protected void Session_End(Object sender, EventArgs e)
    {
    }
    protected void Application_End(Object sender, EventArgs e)
    {
    }
}
}
```

在上面的代码中可以看到，Global.asax 是由 System.Web.HttpApplication 类派生的。它提供了 Application_Start，Application_End，Session_Start，Session_End 等应用程序级事件。在相应的事件中，都添加了一些代码，以使得读者可以清晰地看到 Global.asax 文件的执行情况。

随后就要进行 Web 页面的加工了，主要对 WebForm1.aspx 进行设计。其中，界面设计如图 1-2 所示。

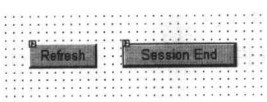

图 1-2　界面设计

在这里添加了两个按钮，分别命名为 Refresh 和 Session End，用于完成刷新和关闭