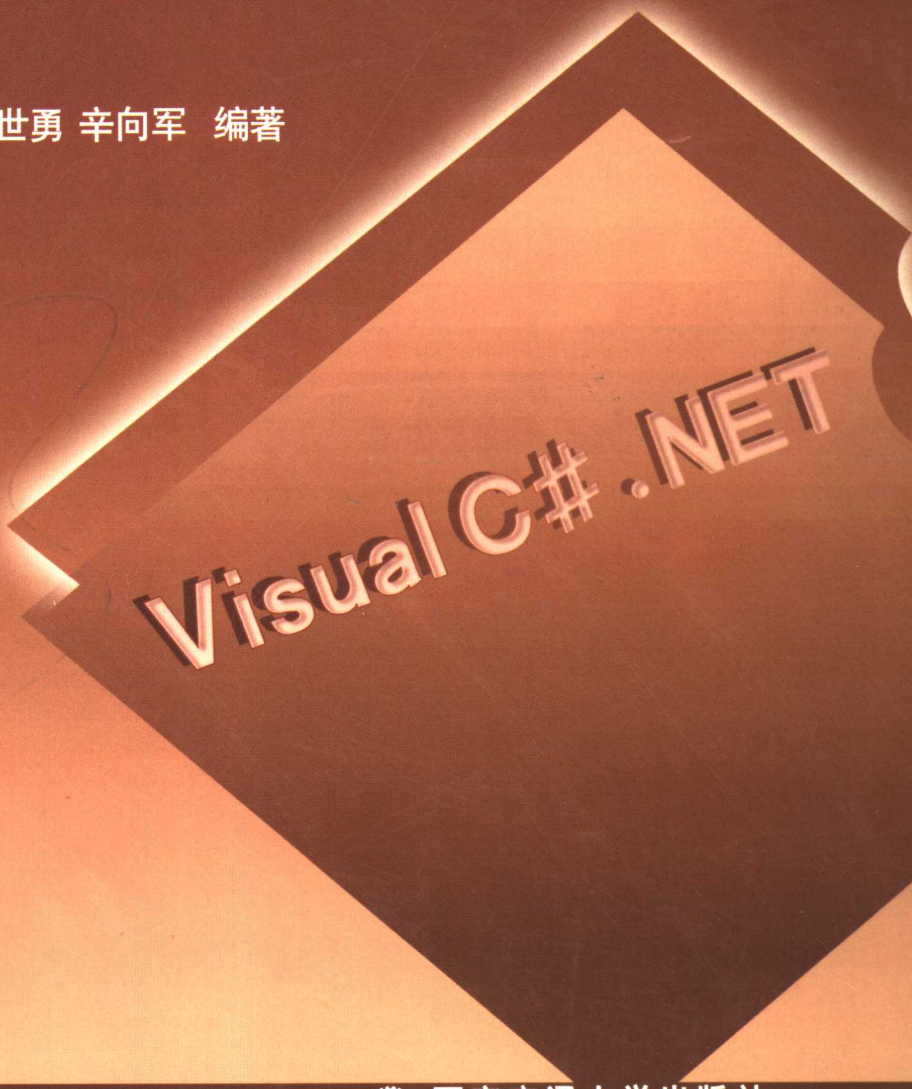
21世纪工程应用计算机技术丛书

Visual C#.NET
与网络数据库编程

李用江 杨世勇 辛向军 编著

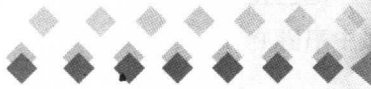


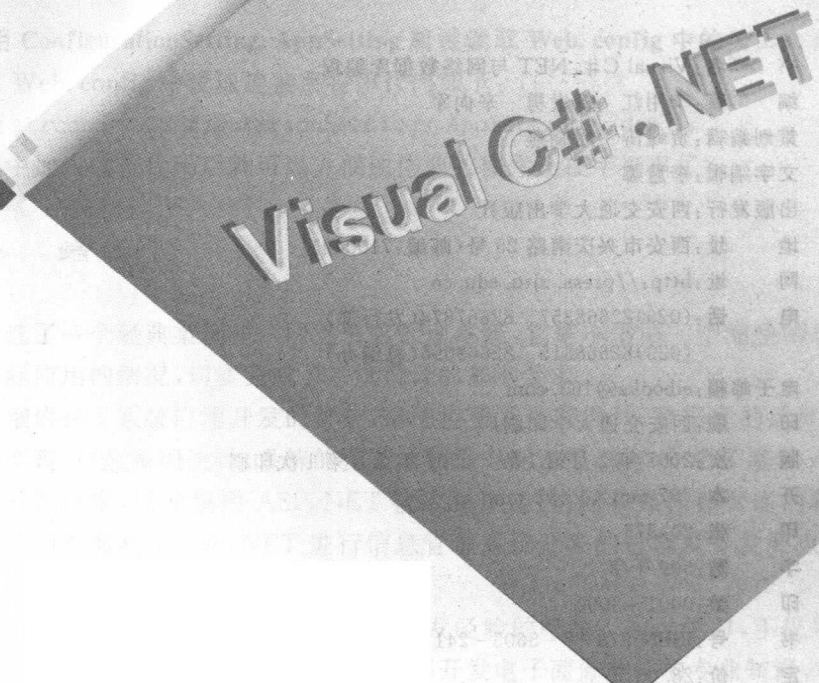
西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS

21世纪工程应用计算机技术丛书

Visual C#.NET
与网络数据库编程

李用江 杨世勇 辛向军 编著





西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS
·西安·

内容提要

本书以C#.NET中文版集成开发环境为基础，将基础语言与数据库应用程序开发紧密结合起来，深入系统地介绍使用C#开发数据库应用程序的各项关键技术。

全书共分8章，所涉及到的内容包括C#语言的基础知识和开发环境、开发Windows窗体和Web窗体的方法和步骤、ADO.NET数据库访问技术的理论基础及其优点、开发基于C/S或B/S模式的数据库应用系统的实现方法，最后通过一个大型开发实例深入地讲述了C#数据库编程技术及技巧，内容涵盖了有关C#数据库编程的方方面面，并且介绍了网络数据库Web发布等新技术。

本书内容全面、讲解透彻、步骤清晰、图文并茂、语言生动流畅、举例典型实用，能够使读者在轻松愉快的环境中快速掌握Visual C#.NET网络数据库编程的方法与技巧。

本书主要是为已经对C#语言有所了解并想开发C#数据库应用程序的读者所编写，并可作为广大编程爱好者和专业程序员的参考资料，另外，它还是大学生做毕业设计的绝佳参考书。

读者在使用本书过程中的技术问题，请发电子邮件至eibooks@163.com与我们联系。本书所需示例程序的源代码请从网址：press.xjtu.edu.cn/xiazai.htm下载，本书所有案例程序均经过调试，可以直接使用。

图书在版编目(CIP)数据

Visual C#.NET与网络数据库编程/李用江等编著.
—西安：西安交通大学出版社，2007.2
21世纪工程应用计算机技术丛书
ISBN 978-7-5605-2417-7

Ⅰ.V... Ⅱ.李... Ⅲ.C语言-程序设计
Ⅳ.TP312

中国版本图书馆CIP数据核字(2007)第002209号

书　　名：Visual C#.NET与网络数据库编程
编　　著：李用江　杨世勇　辛向军
策划编辑：贺峰涛　屈晓燕
文字编辑：李慧娜
出版发行：西安交通大学出版社
地　　址：西安市兴庆南路25号(邮编：710049)
网　　址：http://press.xjtu.edu.cn
电　　话：(029)82668357　82667874(发行部)
　　(029)82668315　82669096(总编办)
电子邮箱：eibooks@163.com
印　　刷：西安交通大学印刷厂
版　　次：2007年2月第1版　2007年2月第1次印刷
开　　本：787 mm×1 092 mm　1/16
印　　张：20.875
字　　数：509千字
印　　数：0001～3000
书　　号：ISBN 978-7-5605-2417-7/TP·489
定　　价：28.00元

前　言

Microsoft. NET 是微软公司推出的面向网络、支持各种用户终端和数据库的、多语言环境的、最新一代开发平台的简称，有人称它将彻底改变软件的开发方式、发行方式和使用方式。. NET 作为第三代 Internet 产品与 Java 有着极大的不同，用户数据生存于网络，而不只是生存于不同平台。. NET 允许不同应用程序之间能互相传递信息，而 Java 的 beans 是不能和其它语言建立的构件共享数据的。. NET 是一个中立语言，基于 XML(Extensible Markup Language)和 SOAP(Simple Object Access Protocol)，允许开发者使用各自不同语言来实现它所有的功能，所以可以肯定，在网络飞速发展的今天，. NET 将很快成为主流。

C#(读作 C-sharp)是微软根据其. NET 战略提出的一种全新的、企业级计算机领域开发. NET 应用的下一代面向对象的首选编程语言，它能够让开发人员在. NET 平台上快速地建立大量的应用程序。按照微软给出的定义，C# 是一种源自于 C 和 C++ 的、简单的、现代的、面向对象的和类型安全的程序设计语言。C# 为程序员提供了开发飞速发展的 Web 应用程序所需的强大而灵活的功能。C# 和 Java 的核心与 C++ 比较有着相同的优势和局限，比起 C++，C# 将更容易被人们理解和接受，现在已经有大量. NET 平台的应用系统由 C# 开发。

ASP. NET 是新一代 Web 应用程序开发平台，它为用户提供了完整的可视化开发环境。本书以 C#. NET 中文版集成开发环境为基础，以结合数据库的 ASP. NET 应用程序开发为重点，介绍了网络应用程序的数据库和 ASP. NET 应用程序的开发方法，以及结合两者在一起的 Web 应用程序的开发方法和关键技术。本书的一个特点就是通过每一个具体的实例来介绍与网络数据库应用程序相关的开发方法，同时全书贯穿以知识的系统性和实用性，为读者提供最为实用的开发技术。

数据库应用是应用软件的根本，全书共分 8 章，都是紧密地围绕着开发数据库应用程序而设计的。第 1 章概述了 C# 程序设计语言的基础知识和开发环境；第 2 章简单介绍了开发 Windows 窗体和 Web 窗体的方法和步骤；第 3 章从理论和实例两个方面深入地讨论了创建 Web 应用程序的方法和技术；第 4 章详细地介绍了 C# 中最重要的数据库访问方法——ADO. NET 的理论基础及其优点；第 5,6 章结合多个案例全面地介绍了开发基于 C/S 模式的数据库应用程序的多种实现方法和数据操纵技术；第 7 章深入地讨论了如何使用 ASP. NET 连接、访问和操作数据库的基本方法，以及使用 ASP. NET 中的服务器控件读取、显示、添加、修改和删除数据库中数据的方法；第 8 章通过一个大型开发实例，从创建一个网络应用程序的数据库和构建 Web 应用程序网站的实用技术出发，深入地讲述了 C# 数据库编程技术及技巧，内容涵盖了有关 C# 数据库编程的方方面面，并介绍了 Internet 网站的创建方法和网络数

据库 Web 发布等新技术，为以后软件开发奠定了坚实的基础。

本书的主要特点是内容全面、讲解透彻、步骤清晰、图文并茂、语言生动流畅、举例典型实用，能够使读者理论学习傻瓜化、案例操作步骤化、开发过程图文化、系统开发实例化，在轻松愉快的环境中快速掌握 Visual C#.NET 网络数据库编程的方法与技巧。

在编写本书的过程中得到了许多老师和同学的帮助和鼓励，如果说没有他们的支持，很难想象一个人如何独自完成如此繁重的任务。这里特别要提到的是广东海洋大学外国语学院的李春长博士，早在 2001 年，那时 Microsoft.NET 刚推出西文测试版，为了学习 C# 这种新的语言，作者和几位学生一起翻译了帮助文档，最后由李春长博士对译文进行了校对，这对大家学习 C# 语言起到了很大的帮助。在近期的写作过程中，常常和西安电子科技大学的计算机学院的赵锟博士交换意见，他对本书的写作提出过许多宝贵的建议，有问题时经常会想和他商讨一下，以便确定大方向是否正确，内容是否恰当，在此表示深深地感谢！在本书的编写过程中，还得到了西安交通大学出版社的贺峰涛先生和屈晓燕女士的热情帮助，他们对本书的内容编排上提出了许多宝贵的意见，使全书的结构更为紧凑，在此向他们致以最衷心的谢意。帮助过我的朋友和老师非常之多，在这里不能一一提到，在此向所有给予鼓励和支持我的同志致以我最衷心的感谢！

本书的第 1 章由郑州轻工业学院信息与计算科学系辛向军博士撰写，第 2 章由西安电子科技大学计算机学院杨世勇博士撰写，第 3～8 章由广东海洋大学信息学院李用江副教授撰写。全书由李用江博士统稿。沈阳化工学院自动控制与检测学院的硕士研究生刘丹斌也参加了部分整理工作和大量程序的调试。

本书得到了陕西省科学技术研究发展计划项目（数学版权管理网络平台的开发，批准号：2005K04－G10）和广东海洋大学科研处的资助，特此致谢！

本书编写及出版过程中，得到了广东海洋大学科研处、信息学院和计算机网络技术广东省重点实验室的大力支持，同时得到了西安电子科技大学计算机学院、ISN 国家重点实验室以及郑州轻工业学院信息与计算科学系的大力支持，在此深表感谢。

由于编写时间较为仓促，书中难免会有疏漏和不足之处，恳请广大读者、专家提出宝贵意见。

作　者

2006 年 12 月

目 录

第 1 章　C#. NET 语言概述

Microsoft. NET(简称 . NET)是 Microsoft 公司推出的面向网络、支持各种用户终端和数据库的、多语言环境的、最新一代开发平台的简称,它包括 Visual Basic. NET、Visual C++ . NET、Visual C# . NET 和 Visual J# . NET 四种编程语言,它们全都使用相同的集成开发环境(IDE),该环境允许它们共享工具并有助于创建混合语言解决方案,它可以用于生成 ASP Web 应用程序、XML Web services、桌面应用程序和移动应用程序。其中 C# . NET(发音为 see-sharp dot net)是微软根据其 . NET 战略提出的一种全新的、企业级计算机领域开发 . NET 应用的首选编程语言,其固有的特性保证了它是一种高效、安全、灵活的现代程序设计语言,使得它能够完成从商业中的组件到系统级程序的一切应用。

1.1　Microsot. NET 简介

现在,大家都已看到所有微软的产品都打上了 . NET 标记。那么究竟什么是 . NET? 我们先要从 Windows DNA(Distribnted Network Architecture)说起。

1.1.1　Windows DNA

在基于 Web 的开发模型中,应用程序是分布式的,也就是说:程序的一些部分运行在客户机上,一些在对象服务器上,另外一些在数据库服务器上。这种结构有很多优点,包括伸缩性、性能、易于配置和扩展,但是它们的有效性要依赖于正确的实现。一个特殊的平台便脱颖而出:Microsoft Windows DNA。Windows DNA 的结构涉及到一个 Web 服务器(Internet Information Server)和一个事务处理服务器(Microsoft Transaction Server),以及其他几个产品和服务。这些服务器经常被单独地称为 IIS 和 MTS。

这一切共同形成了位于 Microsoft Windows 顶部的一个软件平台层,它们开放地紧密集成起来,并提供丰富的应用程序服务。MTS 同 Microsoft Windows 2003 下的强大服务结合在一起,形成了现在大家所知道的 COM+,即:MTS+COM=COM+。

Windows DNA 结构使我们可以把精力集中在最需要的开发领域以满足业务需求,而不用对它的支撑层操心过多。Windows DNA 提供内部结构和服务,方便地实现安全性、事务管理以及同已有系统的互操作性。采用 Windows DNA 结构意味着我们能够一步一步地开始使应用程序支持 Web,而不是从一无所有做起。通过支持最新的可扩展标记语言 XML 和简单

对象访问协议 SOAP 技术，我们可以确信应用程序能够很好地工作于现在和未来。

使用 Windows DNA 结构，我们能够建立可伸缩、可扩展的电子商务解决方案。但电子商务应用程序将如何与其他电子商务程序通讯？这些应用程序究竟在哪里运行？面对这些挑战，Microsoft 亮出了它的最新一代操作平台：Microsoft. NET。

1.1.2 什么是 .NET

.NET 首先是一个开发平台，它定义了一种公用语言子集(Common Language Subset，CLS)，这是一种为符合其规范的语言与类库之间提供无缝集成的混合语。.NET 统一了编程类库，提供了对下一代网络通信标准、可扩展标记语言(Extensible Markup Language，XML)的完全支持，使应用程序的开发变得更容易、更简单。Microsoft. NET 计划还将实现人机交互方面的革命，微软在其软件中添加了手写和语音识别的功能，让人们能够与计算机进行更好的交流，并在此基础上继续扩展功能，增加对各种用户终端的支持能力。最为重要的是 .NET 将改变因特网的行为方式：软件将变成为服务。与 Microsoft 的其他产品一样，.NET 与 Windows平台紧密集成，并且与其他微软产品相比它更进一步：其运行库已经与操作系统融合在了一起，从广义上把它称为一个运行库也不为过。

简而言之，.NET 是一种面向网络、支持各种用户终端的开发平台环境。微软的宏伟目标是让 Microsoft. NET 彻底改变软件的开发方式、发行方式、使用方式等等，并且不止是针对微软一家，而是面向所有开发商与运营商！.NET 的核心内容之一就是要搭建第三代因特网平台，这个网络平台将解决网站之间的协同合作问题，从而最大限度地获取信息。在 .NET 平台上，不同网站之间通过相关的协议联系在一起。网站之间形成自动交流，协同工作，提供最全面的服务。

1.1.3 支持多种编程语言的 Visual Studio. NET

Microsoft Visual Studio. NET 2003 是一套完整的开发工具，用于生成 ASP Web 应用程序、XML Web services、桌面应用程序和移动应用程序。Visual Basic. NET、Visual C++ .NET、Visual C#.NET 和 Visual J#.NET 全都使用相同的集成开发环境(IDE)，该环境允许它们共享工具并有助于创建混合语言解决方案。有关更多信息，请参阅帮助文档中的 .NET 框架概述。另外，这些语言利用了 .NET Framework 的功能，此框架提供对简化 ASP Web 应用程序和 XML Web services 开发的关键技术的访问。

.NET 框架是用于生成、部署和运行 XML Web services 及应用程序的多语言环境。它包含以下三个主要部分。

1. 公共语言运行库

运行库实际上在组件的运行时和开发时的操作中都起作用，尽管名称中没有体现这个意思。在组件运行时，运行库除了负责满足此组件在其他组件上可能具有的依赖项外，还负责管理内存分配、启动和停止线程和进程，以及强制执行安全策略。在开发时，运行库的作用稍有变化，由于做了大量的自动处理工作(如内存管理)，运行库使开发人员的操作非常简单，尤其是与今天的 COM 相比，特别是反射等功能显著减少了开发人员为将业务逻辑转变为可重用组件而必须编写的代码量。

2. 统一编程类

框架为开发人员提供了统一的、面向对象的、分层的且可扩展的类库集(API)。目前，C++开发人员使用 Microsoft 基础类，而 Java 开发人员使用 Windows 基础类。框架统一了这些完全不同的模型，并且为 Visual Basic 和 JScript 程序员同样提供了对类库的访问。通过创建跨所有编程语言的公共 API 集，公共语言运行库使得跨语言继承、错误处理和调试成为可能。从 JScript 到 C++的所有编程语言具有对框架的相似访问，开发人员可以自由选择他们要使用的语言。

3. ASP. NET

ASP. NET 建立在 .NET 框架的编程类之上，它提供了一个 Web 应用程序模型，并且包含使生成 ASP Web 应用程序变得简单的控件集和结构。ASP. NET 包含封装公共 HTML 用户界面元素(如文本框和下拉菜单)的控件集。但这些控件在 Web 服务器上运行，并以 HTML 的形式将它们的用户界面推送到浏览器。在服务器上，这些控件公开一个面向对象的编程模型，为 Web 开发人员提供了面向对象的编程的丰富性。ASP. NET 还提供结构服务(如会话状态管理和进程回收)，进一步减少了开发人员必须编写的代码量并提高了应用程序的可靠性。另外，ASP. NET 使用这些同样的概念使开发人员能够以服务的形式交付软件。使用 XML Web services 功能，ASP. NET 开发人员可以编写自己的业务逻辑并使用 ASP .NET结构通过 SOAP(简单对象访问协议)交付该服务。有关更多信息，请参阅 SOAP 社区链接的相关文献。

1.1.4　面向 .NET 的全新开发工具——C#

在最近的一段时间里，C 和 C++一直是最有生命力的程序设计语言。这两种语言为程序员提供了丰富的功能，高度的灵活性和强大的底层控制能力。而这一切都不得不在效率上做出不同程度的牺牲，又使我们必须要忍受学习的艰苦和开发的长期性，许多 C 和 C++程序员一直在寻求一种新的语言，以图在开发能力和效率之间取得更好的平衡。

也有许多语言为了提高软件生产率，而不得不放弃了一些非常有用的功能，如对底层的控制，这使得它们与先前存在系统很难互操作，而且完成 Web 程序设计实践也相当困难。

对程序员们来说，理想的解决方案是能把快速开发同具有系统平台底层开发能力结合起来，能够把现有的应用和 Web 标准统一。在 .NET 平台上，应用软件的开发将要越来越多地用到 Web 标准，例如超文本链接标示语言(HTML)、可扩展标志语言(XML)以及简单对象访问协议(SOAP)，显然现有的开发工具不可能对 Web 技术提供最好的支持。

为了解决上述问题，微软根据其 .NET 战略提出的一种全新的语言 C#(发音为 C-sharp)，并且在最新版的 Vsual Studio. NET 中提供了 C# 的开发工具。C# 良好的面向对象特性，使得它能够完成从商业中的组件到系统级程序的一切应用。通过 C# 言中简单的构造，这些组件能够很容易地变为 Web 服务，并且允许它们在因特网上被运行任何其他语言的操作系统所调用。

真正使用 C# 语言的原因是：

① C# 是一种精确、简单、类型安全、面向对象的语言，它使企业程序员得以构建广泛的应用程序。

② C# 还凭借以下功能，为软件员提供生成持久系统级组件的能力：

- 对集成现有代码提供完全 COM/平台支持。
- 通过提供垃圾回收和类型安全实现可靠性。
- 通过提供内部代码信任机制保证安全性。
- 完全支持可扩展元数据概念。

③ C#还可以凭借以下功能，与其他语言交互操作、跨平台互用并与遗留的数据交互操作：

- 通过 COM+ 1.0 和 .NET 框架服务提供具有紧密库访问的完全相互作用支持。
- 对基于 Web 的组件交互提供 XML 支持。
- 版本转换功能使管理和部署变得简单。

在当今的网络经济中，速度的竞争显得越来越重要，开发人员不断被要求加快开发速度，甚至要为同一种产品开发多种版本。C#在设计时考虑到了这种需求，它可以帮助开发人员写更少的代码行并减少犯错误的次数。

以上种种都显示了 C#之所以能成为一门热门语言，与其自身的诸多优秀特性是分不开的。

1.1.5 C#语言的特点

在我们已经能用 C++或 Visual Basic 来开发企业应用的情况下，为什么还需要学习另一种编程语言？从市场角度考虑，C#将成为企业级计算机领域开发 .NET 应用的首选语言。

作为编程语言，C#在带来对应用程序的快速开发能力的同时，并没有牺牲 C 与 C++程序员所关心的各种特性，它忠实地继承了 C 和 C++的优点。如果你对 C 或 C++有所了解，就会发现它是那样的熟悉。即使是一位新手，C#也不会给你带来任何其他的麻烦，快速应用程序开发(Rapid Application Development，RAD)的思想与简洁的语法将会使你迅速成为一名熟练的开发人员。

正如前文所述，C#是专门为 .NET 应用而开发出的语言。这从根本上保证了 C#与 .NET 框架的完美结合。在 .NET 运行库的支持下，.NET 框架的各种优点在 C#中表现得淋漓尽致。让我们先来看看 C#的一些突出的特点，相信在以后的学习过程中，你将会深深体会到“#”——“Sharp”的真正含义。

1. 简单性

你一定不认为 C++具有简单易学的特点，而 C#语言却不一样。这种编程语言的首要目标就是简单性，它通过增加许多特性，同时舍弃了一些特性，对 C#的总体简单性作出了贡献。

在缺省的情况下，C#的代码在 .NET 框架提供的“可操控”环境下运行，不允许直接地内存操作。它所带来的最大特色是没有了指针。与此相关的，那些在 C++中被疯狂使用的操作符(例如：“::”“->”和“.”)已经不再出现。C#只支持一个“.”(即“dot”)，对于我们来说，现在需要理解的一切仅仅是名字嵌套而已。

C#用真正的关键字换掉了那些把活动模板库(Active Template Library，ALT)和 COM 搞得乱糟糟的伪关键字，如 OLE_Color、BOOL、VARIANT_BOOL、DISPID_XXXXX 等等。每种 C#类型在 .NET 类库中都有了新名字。

你不再需要记住那些源于不同处理器结构的神秘类型，包括可变长的整数类型，C#通过

提供一个统一的类型系统去掉了这些问题。这个类型系统使你可以将每种类型看作一个对象，不管它是初始数据类型还是发育完全的类。与别的语言不同的是，把单个类型当作对象看待并不会增加执行上的难度，因为它提供了一个叫做装箱(boxing)和拆箱(unboxing)的机制。基本的一点是，这个技术仅在需要的情况下才让一个对象去访问单个类型。

C# 也去掉了多年来在 C++ 中蔓延的冗余问题，比如 const 和 #define、各种的字符类型等等。常用的形式在 C# 中被保留下来，而别的冗余形式从 C# 语言中被清除出去了。

2. 现代性

.NET 运行库提供了代码访问安全特性，它允许管理员和用户根据代码的 ID 来配置安全等级。在缺省情况下，从 Internet 和 Intranet 下载的代码都不允许访问任何本地文件和资源。比方说，一个在网络上的共享目录中运行的程序，如果它要访问本地的一些资源，那么异常将被触发，它将会被异常无情地扔出去，若拷贝到本地硬盘上运行，则一切正常。内存管理中的垃圾收集机制减轻了开发人员对内存管理的负担。NET 平台提供的垃圾收集器(Garbage Collection，GC)将负责资源的释放与对象撤销时的内存清理工作。

变量是类型安全的。C# 中不能使用未初始化的变量，对象的成员变量由编译器负责将其置为零，当局部变量未经初始化而被使用时，编译器将做出提醒；C# 不支持不安全的指向，不能将整数指向引用类型，例如对象，当进行下行指向时，C# 将自动验证指向的有效性；C# 中提供了边界检查与溢出检查功能。

借鉴了 SQL 等语言，C# 增加了诸如小数和字符串等基本数据类型，C# 也开始采用更现代化的调试方法。

3. 面向对象

C++ 也是面向对象的，但使用过 C++ 的读者一定不会忘记在用到多重继承时遇到的问题。C# 为了更好地支持 COM+ 的虚拟对象系统隐藏了多重继承，而封装、多态和继承被保留下来。C# 只允许单继承，即一个类不会有多个基类，从而避免了类型定义的混乱。C# 隐藏了全局函数、变量和实例，作为替代，能够创造静态类成员。C++ 中，可以通过创造一个类成员并且能在后面重新定义该方法。

C# 中方法的默认属性是非虚拟的，相应地提供了一个虚拟修饰符，可以更容易地进行版本控制。类成员在 C# 中的存取属性值可以为私有(Private)、受保护(Protected)、公共(Public)或者内在(Internal)。

借助于从 VB 中得来的丰富的 RAD 经验，C# 具备了良好的开发环境。结合自身强大的面向对象功能，C# 使得开发人员的生产效率得到极大的提高。对于公司而言，软件开发周期的缩短将能使他们更好地应付网络经济的竞争。

4. 类型安全性

C# 实施了最严格的类型安全来保护它自身及其垃圾收集器。因此，在 C# 中必须遵守关于变量的一些规定：

- 不能使用未初始化变量。对于对象的成员变量，编译器负责将它们置零。局部变量应自己负责。如果使用了未经初始化的变量，编译器会进行提醒。这样做的好处是：可以摆脱因使用未初始化变量得到一个可笑结果的错误。
- C# 不支持不安全的指向。不能将整数指向引用类型(比如对象)，当进行下行指向时，

C# 会验证指向的有效性(就是说,导出对象确实是从用户要将它下行指向的类型中导出的)。

· 边界检查是 C# 的一部分。当数组实际上只有 $n-1$ 个元素时,不可能使用它"额外"的数组元素,这使重写未经分配的内存成为不可能。

· 算术运算可能溢出结果数据类型的范围。C# 允许在应用级或者语句级检查这类操作中的溢出,使用溢出检查,当溢出发生时会出现一个异常。

· C# 中传递的引用参数是类型安全的。

5. 版本处理技术

在过去的几年中,几乎所有的程序员都至少一次地和已成为所谓的"DLL 地狱"打过交道,产生这个问题是因为许多计算机上安装了同一 DLL 的不同版本。有时,旧版的应用在较新版的 DLL 中有幸还可以运行,但大多数情况下,它们崩溃了。版本处理技术是当今令我们真正很痛苦的一个问题。

C# 提供内置的版本支持来减少开发费用,使用 C# 将会使开发人员更加轻易地开发和维护各种商业应用。

升级软件系统中的组件(模块)是一件容易产生错误的工作。在代码修改过程中可能对现存的软件产生影响,很有可能导致程序的崩溃。为了帮助开发人员处理这些问题,C# 在语言中内置了版本控制功能。例如:函数重载必须被显式地声明,而不会像在 C++ 或 Java 中经常发生的那样不经意地被进行,这可以防止代码级错误并保留版本化的特性。另一个相关的特性是接口和接口继承的支持。这些特性可以保证复杂的软件可以被方便地开发和升级。

6. 兼容性

C# 允许与 C 风格所需要传递指针型参数的 API 进行交互操作,DLL 的任何入口点都可以在程序中进行访问。C# 遵守 .NET 公用语言规范 CLS,从而保证了 C# 组件与其他语言组件间的互操作性,C# 能够透明访问 COM 和 OLE 的 API,并且不用考虑细节问题,还可以通过 COM+ 的运行语言支持其所有的数据类型。

7. 灵活性

在简化语法的同时,C# 并没有失去灵活性。尽管它不是一种无限制的语言,比如:它不能用来开发硬件驱动程序,在默认的状态下没有指针等等,但是,在学习过程中我们将发现它仍然是那样的灵巧。

如果需要,C# 允许将某些类或者类的某些方法声明为非安全的。这样一来,我们将能够使用指针、结构和静态数组,并且调用这些非安全的代码不会带来任何其他的问题。此外,它还提供了 delegates(代表)来模拟指针的功能;C# 不支持类的多继承,但是通过对接口的继承,将获得这一功能。

8. Web 性

.NET 中新的应用程序开发模型意味着越来越多的解决方案需要与 Web 标准相统一,例如超文本标记语言(Hypertext Markup Language,HTML)和 XML。由于历史的原因,现存的一些开发工具不能与 Web 紧密地结合。SOAP 的使用使得 C# 克服了这一缺陷,大规模深层次的分布式开发从此成为可能。

由于有了 Web 服务框架的帮助,对程序员来说,网络服务看起来就像是 C# 的本地对象。程序员们能够利用他们已有的面向对象的知识与技巧开发 Web 服务。仅需要使用简单的

C#语言结构，C#组件将能够方便地为 Web 服务，并允许它们通过 Internet 被运行在任何操作系统上的任何语言所调用。举个例子，XML 已经成为网络中数据结构传送的标准，为了提高效率，C#允许直接将 XML 数据映射成为结构。这样就可以有效地处理各种数据。

9. 可伸缩性

在 C 和 C++语言编译之前，都需要各种经常不相容的头文件。C#通过组合类型的声明和定义舍弃了这些繁琐的头文件，它也可以直接输入输出 COM+的元数据，使编译更容易。

当设计一个特别大的项目时，可能想把代码分成几个较小的源文件，而且编译项目时，可以把这些源文件看成已经串连起来的一个大的文件。这意味着在编译时，不必考虑哪个例程属于哪个源文件，开发者甚至可以移动、重命名、分割和删除它们而不必中断编译。

总之，C#语言是从 C、C++导出的，它是为了那些愿意牺牲 C++某些原有能力换得更多便利和效率的企业应用开发者而创造的。C#是现代的、面向对象的语言，它使开发者快速、简单地为微软 .NET 平台建立解决方案。它所提供的框架允许 C#组件变成 Internet 上运行在任何平台上的任何应用都可获得的 Web 服务。C#语言提高了开发者的生产力，同时，消除了可能导致开发成本增加的编程错误。C#语言为 C/C++开发者提供了快速的 Web 开发环境，同时也保留了 C/C++程序员想要的功能和灵活性。我们相信在不远的将来 C#语言会成为软件开发的主要工具。

1.2　Visual C#.NET 集成开发环境介绍

接下来我们将介绍如何安装和使用 Visual C#.NET 的集成开发环境。

1.2.1　Visual Studio.NET 安装

安装 Visual Studio.NET 时，可以在若干不同的安装和设置选项中进行选择。还可以参考位于安装光盘(CD 或 DVD)根目录下的自述文件。自述文件包含有关 Visual Studio.NET 中所有产品的安装问题的详细信息。自述文件采用 HTML 格式，并可以使用 Internet 浏览器(如 Microsoft Internet Explorer 4.0 版或更高版本)进行查看。

1. Visual Studio.NET 硬、软件要求

(1) Visual Studio.NET 硬件要求

现以 Visual Studio.NET 企业级结构设计版本为例，说明对硬件的要求，其他各种版本的硬件要求和 Visual Studio.NET 企业级结构设计版本基本一样。

安装 Visual Studio.NET 企业级结构设计版本的计算机应满足表 1-1 所列系统要求。

表 1-1　系统要求

要　求	企业级结构设计版本
处理器	具有 Pentium Ⅱ级处理器的计算机，450 MHz(建议：Pentium Ⅲ级，600MHz)
RAM	建议：Windows 2000 Professional——128 MB；Windows 2003 Server——256 MB
硬盘空间	系统驱动器上有 900 MB，安装驱动器上有 4.1 GB
操作系统	Windows 2000、Windows XP、Windows Server 2003 或 Windows NT 4.0

(2) Visual Studio. NET 软件要求

Visual Studio 中的某些项目类型和功能要求必须在安装某些特定软件组件(可能在安装中作为可选组件列出)之后,才能使用这些功能或项目。这些组件中有些必须安装在开发计算机上,而有些则可以安装在远程计算机上。

表 1-2 列出了必须在不同的操作系统上安装以执行指定任务的组件。这些组件不用 Visual Studio. NET Windows 组件更新安装。

表 1-2 各操作系统上安装的组件

希 望	Windows NT 4.0	Windows 2003	Windows XP
开发 ASP Web 应用程序和 XML Web services	IIS 4.0,随 Windows NT 4.0 Option Pack 提供	Internet 信息服务(IIS)	IIS
编译与 Microsoft Windows 消息队列(MSMQ)相关的代码	Microsoft Message Queue Server 1.0,随 Windows NT 4.0 Option Pack 提供	消息队列服务	消息队列服务
在远程计算机上调试代码	Visual Studio 远程调试器	Visual Studio 远程调试器	Visual Studio 远程调试器
使用源代码管理来控制存储过程的版本	Visual Studio 6.0 存储过程版本控制 Visual Source Safe Microsoft SQL Server	Visual Studio 6.0 存储过程版本控制 Visual SourceSafe Microsoft SQL Server	Visual Studio 6.0 存储过程版本控制 Visual SourceSafe Microsoft SQL Server

2. Visual Studio. NET 的安装步骤

本安装实例是在中文 Windows Advance 2003 中安装 Visual Studio. NET 2003,依次需要插入以下光盘:CD-A,CD-C,CD-A,CD-B,CD-D,CD-E,CD-F 等。

① 关闭所有打开的应用程序,以防止在安装过程中需要进行系统的重新启动。

② 插入标为 Visual Studio. NET CD-A 或 DVD 光盘。

③ 自动运行会启动 Setup. exe。如果自动运行没有启动的话,则请从安装 CD 或 DVD 的根目录下直接运行 Setup. exe,出现如图 1-1 所示的安装界面。

☛**注意** 如果在安装程序运行的同时正在运行防病毒程序(例如 Norton AntiVirus),则可能会显示警告,这是因为安装程序运行了访问文件系统对象的脚本。允许脚本运行是安全的。另外,Microsoft Office 软件需要重新安装,否则不可运行。

④ 该安装程序对磁盘上已安装的组件进行扫描。如果安装程序确定系统需要进行组件更新,则在"安装"对话框中会出现步骤 1:"安装 Visual Studio. NET 系统必备"。选择步骤 1 来安装 Visual Studio. NET 系统必备,完成后如图 1-2 所示。如果不需要进行组件更新,则该选项会显示为灰色,此时可以直接进入到安装程序的步骤 2。

⑤ 在安装程序确认系统已经包含有系统必备组件之后,"安装"对话框将会启用步骤 2:"安装 Visual Studio. NET"。如图 1-3 所示。

⑥ 如果要开发分布式应用数据库程序,还要安装 MSDN Library,"安装"对话框将会启用步骤 3:"产品文档",如图 1-4 所示。