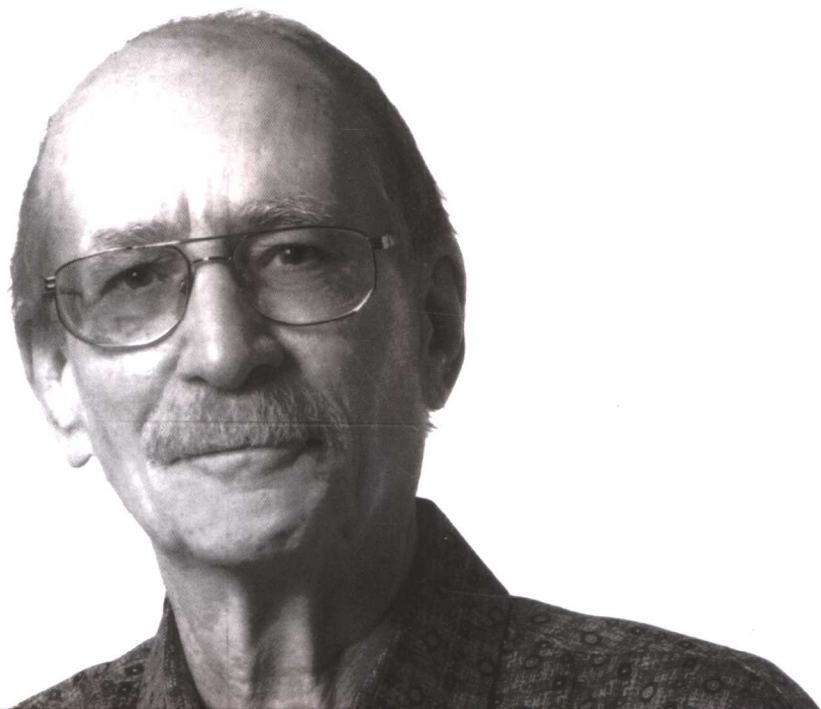




Roger Jennings, OakLeaf Systems 的负责人、Fawcette Technical Publications 网站负责人, 还是 Visual Studio Magazine 的资深编辑, 编著过二十多本微软数据库平台和 Windows 操作系统方面的图书。

expert one-on-one Visual Basic 2005
Database Programming

Visual Basic 2005 数据库专家编程

(美) Roger Jennings 著
沈晓春 崔洪斌 译



清华大学出版社

Visual Basic 2005 数据库专家编程

(美) Roger Jennings 著

沈晓春 崔洪斌 译

清华大学出版社

北 京

Roger Jennings

Expert One-on-One™ Visual Basic® 2005 Database Programming

EISBN: 0-7645-7678-X

Copyright© 2006 by Wiley Publishing, Inc., Indianapolis, Indiana

All Rights Reserved. Authorized translation from the English language edition published by Wiley Publishing, Inc.

本书中文简体字版由 Wiley Publishing, Inc. 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。

北京市版权局著作权合同登记号 图字: 01-2006-1582

版权所有, 翻印必究。举报电话: 010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

本书防伪标签采用特殊防伪技术, 用户可通过在图案表面涂抹清水, 图案消失, 水干后图案复现; 或将表面膜揭下, 放在白纸上用彩笔涂抹, 图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

Visual Basic 2005 数据库专家编程/(美) 詹尼斯(Jennings, R.) 著; 沈晓春, 崔洪斌 译.

—北京: 清华大学出版社, 2006.11

书名原文: Expert one-on-one™ Visual Basic® 2005 Database Programming

ISBN 7-302-13757-9

I. V… II. ①詹…②沈…③崔… III. BASIC 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 105274 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦
http://www.tup.com.cn 邮 编: 100084
社 总 机: 010-62770175 客 户 服 务: 010-62776969

组稿编辑: 王 军

文稿编辑: 徐燕萍

封面设计: 康 博

版式设计: 康 博

印 刷 者: 清华大学印刷厂

装 订 者: 北京市密云县京文制本装订厂

发 行 者: 新华书店总店北京发行所

开 本: 185 × 260 印张: 34.5 字数: 840 千字

版 次: 2006 年 11 月第 1 版 2006 年 11 月第 1 次印刷

书 号: ISBN 7-302-13757-9/TP · 8277

印 数: 1 ~ 4000

定 价: 68.00 元

作者简介

Roger Jennings 先生是 Microsoft Visual Basic .NET n 层和客户端/服务器数据库应用程序以及数据密集型 ASP.NET Web 服务的顾问和作者，他还是软件和系统测试组的一名成员，Roger Jennings 先生参加的测试工作有：从 Professional Extensions for Visual Basic 2.0 (code-named Rawhide)和 Visual Studio 开始的 Visual Basic 的各 beta 版；从 4.2 for OS/2 开始的所有 Microsoft SQL Server 版本；所有 Microsoft Access 版本以及从 Windows 3.0 Multimedia Extensions 开始的所有 Windows 客户端和服务操作系统。

Roger 先生编写的 25 部计算机方面的书籍(英文版)已印刷 125 万册，并被翻译成 20 多种语言。他是《Database Developer's Guide to Visual Basic》(SAMS Publishing)的 3 个版本的作者；《Access Developer's Guide》(SAMS Publishing)的 2 个版本的作者；《Special Edition Using Microsoft Access》(QUE Publishing)的 9 个版本的作者；《Special Edition Using Windows NT 4.0 Server》(QUE)的 2 个版本的作者。他还编写了面向开发员的众多书籍，涉及的内容有 Windows 3.1 多媒体、Windows 95、Windows 2000 Server、Active Directory Group Policy、Visual Basic .NET Web 服务以及 Microsoft Office InfoPath 2003 SP-1。Roger 先生是由 Fawcette Technical Publications 主办的《Visual Studio Magazine》的特约编辑，还是由 Fawcette 主办的《.NETInsight》和《XML & Web Services Insight》时事通讯的专栏作家。

Roger 先生已经有 25 年多的计算机阅历。他最初从事于由 Wang 700 计算器/计算机驱动的实时医学数据获取和化学过程控制系统。Roger 先生是 OakLeaf Systems(北加利福尼亚软件咨询机构)的主管；OakLeaf XML Web Services 站点开发员；并且是 OakLeaf Systems 博克(oakleafblog.blogspot.com)的作者。他的 OakLeaf Code of Federal Regulations (CFR) ASP.NET Web 服务和客户端(www.oakleaf.ws/cfr/)项目荣获 charter Microsoft .NET Best Award for Horizontal Solutions (www.microsoft.com/presspass/features/2002/aug02/08-07netwinners.msp)。Roger 先生的电子邮箱是：Roger_Jennings@compuserve.com。

前 言

自从 Microsoft 于 1998 年 6 月 15 日推出对应的产品后,几乎 80%以上的 Visual Basic 6.0 项目涉及到与一个或多个关系数据库的连接。Access 1.0 于 1992 年引入了连接性引擎技术 (Joint Engine Technology, Jet)索引的顺序访问方法 (ISAM)数据库和 Embedded Basic。Embedded Basic 是 Visual Basic for Applications (VBA)的前身。Access 1.0 和 1.1 能够对 Jet、SQL Server 4.2 以及有开放式数据库连接 (Open Database Connectivity, ODBC) 驱动程序的其他客户端/服务器前端进行快速应用程序开发。Visual Basic 3.0 于 1993 年引入了数据绑定和第一个数据绑定表格控件。随着它们的出现,介绍用 SQL 和 ODBC VB 数据库编程的书籍、杂志和文章像潮水一样涌出。Visual Basic 4.0 引入了 32 位项目,但它并不是一个健壮的开发平台。1995 年推出的 Visual Basic 5.0 增加了动态数据对象 (ActiveX Data Objects, ADO) 1.0 和 OLE DB。VB 和 VBA 当时成为世界上最流行的编程语言,估计用户已达到 3 百万。到 1998 年,大多数专业 VB6 开发员在为客户端/服务器编写产品级 Windows 前端,并开始对三层、分布式数据访问体系结构采用 Microsoft Transaction Server 1.0。在后来的 4 年中,开发人员编写了数以百万计行的数据密集型 VB 代码。

随着 2002 年 Visual Studio .NET 的推出,对基于 COM 的 VB 和 VBA 的升级敲响了丧钟。Visual Basic 6.0 已经移植到从过去的开发平台中保留下来的维护模式的应用中。微软宣布:到 2005 年 3 月 31 日将结束对 VB6 的主流支持,并在 2008 年结束对 VB6 的扩展(付费)支持。2005 年 3 月初,微软最有价值专家 (Microsoft Most-Valued Professionals, MVPs) 组组织了请愿:“……在 Visual Studio IDE 中要包括 VB6 的升级版”。到 3 月中旬,已有 200 多家以前以及当时的 MVPs 对请愿签了名。计算机出版业参加请愿导致了大量博客的出现,它们支持或者反对 VB 或 VBA 以未托管的 VB.COM 形式存在。微软最终采用请愿者建议的可能性很小,而且可能性为零。微软预测将来会出现托管代码。对于 VB 编程人员来说,如果他们不更新其技术,将会被排斥在应用程序生命周期维护之外,或者更糟。如果大家还没有对新的数据密集型项目采用 .NET Framework 和托管 VB 代码的话,那么通过学习本书内容并参考其示例代码和项目就可以成为 VB 2005 数据库的熟练编程人员。

VB 开发人员面临的另一个问题是:与 (Visual) C# 开发人员相比,他们给人以“二等公民”的感觉。由于开发人员对 C# 或 Java 的抵制,要减少 VB 应用的谣言被夸大。VB 2005 比传统的、易读的 VB 语法更全面、兼容性更好。VS 2002/2003 的 VB 语言缺少许多 C# 的语言特征,而且由 Redmond 提供的大多数托管代码示例是用 C# 编写的。Microsoft 开发人员对 C# 的偏爱可以理解,因为大多数开发人员有多年的 C++ 和 JScript 编程经验。VB 2005 通过增加部分类、操作符重载、泛型、数学运算、用于无符号 Integer 和 Long 数据类型的类型转换、用于内部对象处理的 using 关键词以及 XML 注释,使其与 C# 的地位逐渐接近。VB 2005 编程人员几乎可以使用所有 C# 编程结构,但是 C# 缺少 VB 2005 的新的 My 命名空间和已有的 With...End With 结构的便捷性。本书提供了一些详细示例,这些示例可说明适用于数据密集型应用程序的 VB 2005 语言的新结构和新功能。

Visual Studio 和 Visual Basic Express Edition 2005(VBX)利用 Data Sources 窗口增强了作为部分类型的类型化 DataSet 对象的拖放生成功能,还增强了用于 DataAdapter 对象的 TableAdapter 包装器以及 TableAdapter Configuration Wizard。通过从 Data Sources 窗口向 Windows 窗体拖放表图标,可自动添加用户选择的绑定文本框或 DataGridView 控件。DataGridView 控件替换了倍受争议的 GridView 控件。BindingSource 对象将 DataSource 连接到绑定控件,并 BindingNavigator(预置 ToolStrip 控件)处理行导航。利用 Web 窗体,能够通过从 Server 管理器拖表图标的方式生成新的 GridView 和其他数据绑定控件。新的 DataSet 设计器代替了 VS2002/2003 的 XML 架构。本书第 1、2、7 章将介绍如何通过示例数据库,用不到 5 分钟的时间,且不用编写任何代码就能够构建可用的主、子 Windows 窗体和 Web 窗体。其余各章将介绍如何通过事件处理代码自定义或扩展绑定和未绑定控件的功能。

Visual Studio 2005 提供的联机帮助的不足之处是:缺少用于数据密集型项目的复杂 Windows 窗体和 Web 窗体代码示例。帮助文件中的大多数示例代码要生成控制台项目,它们很难用于数据密集型应用程序。读者找不到用于本书、超过 100 MB(压缩后 20 MB)、可下载且以数据为中心的示例项目这样的单控制台项目。各项目示例用 Windows 和 Web 窗体模拟了从简单到中等复杂程度的实际数据库前端应用程序。

本书读者对象

本书面向从 VB6 或 VS 2002/2003 升级到 VB 2005 且有一定编程经验的 VB 编程人员。本书假设读者对 VB 2005 集成开发环境(IDE)有基本的了解,但并不需要有 VB6、VBA 或 VBScript 数据库编程经验(第 1 章“从 ADO 向 ADO.NET 移植”除外)。读者最好编写过 Microsoft Transact-SQL (T-SQL)语句和简单存储过程。如果读者熟悉 XML 1.0 和 XML,则会更易于掌握 DataSets 和 SQL Server 2005 的新 xml 数据类型。如果读者编写过 XPath 1.0 表达式,则有助于对第 12 章(分析 xml 数据类型)的学习。

本书主要内容

本书将重点介绍对 .NET Framework 2.0 的 System.Data 命名空间的编程,该编程可实现 ADO.NET 2.0 和相关的命名空间,如 System.Transactions 和 System.Xml。本书并不是用于 .NET Framework、VS 2005、VBX 或 VB 2005 的新用户指南,而是重点介绍与数据相关的知识。

本书前几章的许多代码和项目示例是直接自定义 SqlConnection、SqlCommand 和 SqlDataReader 对象检索和更新表数据。后面各章的例子使用了由 TableAdapters 填充的强类型化 DataSet,并通过绑定文本框、DataGridView 以及下拉列表显示数据。

SQL Server 2000、MSDE 2000、SQL Server 2005 或 SQL Server 2005 Express (SQLX)可用于本书第 I 部分到第 III 部分各代码和示例项目的数据源。通过修改这 3 部分的大多数例子,可以利用本地 OleDbproviders 运行 Access 2000 文件或更高版本的.mdb 文件。本书

第 IV 部分各章的 T-SQL 和 VB 2005 代码需要 SQL Server 2005 或 SQLX。SQLX 没有包括 SQL Server 2005 的某些新函数，因此读者学习第 IV 部分时，需要使用 SQL Server 2005 Developer Edition 或更高的版本。

本书结构

本书共分 4 大部分，每部分有 3 章。书中的大部分章节都是假设读者已掌握前面章节介绍的知识。因此，建议读者按顺序学习本书。下面简要介绍本书各部分以及各章的主要内容。

第 I 部分：ADO.NET 2.0 基础知识

第 I 部分各章将概要介绍 ADO.NET 2.0，读者对象是转移到 VS 2005 的 VB6 开发人员、或采用 VS 2005 的新数据关联功能的 VBX 和 VS2003/2004 开发人员。

第 1 章：从 ADO 移植到 ADO.NET。介绍 .NET Framework 2.0 的 System.Data 命名空间的成员，并介绍基于 COM 的 ADO 编程与 ADO.NET 2.0 数据对象的 VB6 和 VB 2005 代码的相似与不同之处。本章最后将给出一个例子，说明通过拖放的方式生成一个主-明细窗体，且该窗体的数据源是类型化 DataSet 对象。

第 2 章：ADO.NET 2.0 新功能简介。介绍如何编写新的 ADO.NET 2.0 运行时对象，如 DbProviderFactories 和异步 SqlCommand 对象。本章还介绍如何根据在相关表中更新数据的类型化 DataSets 创建复杂窗体。

第 3 章：对以数据为中心的项目采用最佳实践。首先介绍微软对分层最佳实践的推荐标准：模式与实践白皮书、设计指南以及用于数据密集型和面向服务的 .NET 项目的应用程序代码块库。本章通过实现示例总结出 ADO.NET 2.0 项目的推荐标准。

第 II 部分：Windows 窗体和控件中的数据绑定

VS 2005 的新的 ClickOnce 开发功能允许用户部署自更新的 Windows 窗体应用程序，用户能够通过 Web 站点安装、更新和运行这些应用程序。ClickOnce 部署可以使开发人员用智能 Windows 窗体客户端代替基于 Web 的应用程序。智能客户端为用户提供了多种形式和响应式 UI、传输离线操作，并提高了数据安全性。第 II 部分各章介绍用户如何以类型化 DataSets 作为他们的数据源来设计和编写智能客户端。

第 4 章：对 TableAdapters、BindingSources 和 DataGridViews 编程。介绍如何用 FillBy 方法参数化主-明细-子明细窗体、格式化 DataGridViews 并对其添加计算列、提供默认值、向 DataGridView 列添加查询组合框以及如何在 3 个相关表中更新数据。

第 5 章：添加数据验证和并发管理。介绍如何为绑定文本框和 DataGridViews 编写数据验证代码、合理地管理并发冲突以及如何利用本地持久式类型化 DataSet 接受非连接用户。

第 6 章：高级 DataSet 技术。介绍如何利用 DataSet 部分类的优势、为事物管理编写部分类代码、在 DataGridViews 中显示和处理图像、通过 XML 文档及其架构生成 DataSet、将序列化对象作为数据源以及如何将 DataGridViews 绑定到通用 BindingList 集。

第 III 部分：ASP.NET 2.0 中的数据绑定

ASP.NET 2.0 与 ASP.NET 1.x 有很大的不同。在 ASP.NET 2.0 中，已不存在基于表格、位置固定的布局，而是提供了新的 Visual Web Developer，而且轻量级 Web 服务可提高开发速度。数据绑定 GridView 控件代替了 DataGrids，新的 FormView 和 DetailsView 控件简化了数据密集型 Web 应用程序的创建过程。

第 7 章：ASP.NET 2.0 的 DataSource 和绑定控件。介绍 ASP.NET 2.0 的新设计器以及基于流的布局的代码。本章介绍如何为绑定控件生成 SqlDataSources、为数据绑定控件创建模板以及如何添加 DataList、DropDownList、FormView、GridView 和 DetailsView 控件。

第 8 章：高级 ASP.NET 2.0 数据技术。介绍数据验证、对象和 XML 数据源；页面级和应用程序级的性能跟踪以及利用复制文件或编译的 DLL 部署到 IIS Web 站点。

第 9 章：发布数据驱动的 Web 服务。本章可使读者生成默认的“Hello World”Web 服务及其客户端代理，然后为 Web 服务身份验证添加客户端凭证。数据密集型 Web 服务可说明如何以 SOAP 响应消息的方式向 Windows 窗体 Web 服务客户端返回类型化 DataSet，并更新服务的表。本章的最后一个例子是向 Web 窗体客户端返回自定义业务对象的 Web 服务。Web 窗体客户端的绑定数据控件要更新 Web 服务的对象数据源。

第 IV 部分：SQL Server 2005 和 ADO.NET 2.0

SQL Server 2005 对 SQL Server 2000 进行了大的升级，这一点可通过长达 5 年的升级周期来证明。在第 IV 部分中，各章将重点介绍新的和改进的 Server 2005 以及 SQLX 功能、T-SQL 扩展、VB 2005 SQL Server (SQLCLR)项目和新的 xml 数据类型。

第 10 章：从 SQL Server 2000 升级到 SQL Server 2005。介绍 SQL Server 2005 各版本之间的区别，并介绍如何使用新的管理工具和报表服务。本章将详细介绍新的数据库引擎功能(如服务代理、数据库邮件、查询通知以及本地 SOAP Web 服务)示例，还会介绍新的 T-SQL 关键词，如 PIVOT/UNPIVOT、FOR XML RAW、PATH 以及 TYPE 修改符。

第 11 章：创建 SQL Server 项目。通过学习本章，读者应该可以为托管存储过程和触发器以及自定义类型、函数、聚合和触发器创建和部署 VB 2005 SQLCLR 项目。

SQL Server 2005 Developer Edition 或更高的版本提供了自动部署 SQLCLR 项目时需要的 SQL Server Project 模板。Visual Basic Express Edition 没有包括 SQL Server Project 模板。

第 12 章：xml 数据类型。介绍本地 xml 数据类型，并介绍如何用 XML 架构创建强类型化 xml 列、添加索引、如何编写 XQuery 表达式来返回标量值或节点集，以及如何在 xml 列更新文档实例的元素或属性。

学习本书需要做的工作

读者的计算机必须满足使用 VS 2005 版本(或 VBX)和 SQL Server 2005(或 SQLX)对系统的最小需求。Microsoft 对 SQL Server 2005 的最低硬件要求是: Intel 或兼容 Pentium III 处理器、速度为 550 MHz 以上、256 MB 的 RAM、硬盘容量最小为 1 GB RAM。

如果读者用低于 833 MHz 的处理器和小于 512 MB 的 RAM 运行 VS 2005 或 VBX, 开发过程受到一定的限制。

安装全部 SQL Server 2005 Developer Edition 或更高的版本时, 大约需要 1.6 GB 的硬盘空间, 其中 200 MB 用于 .NET Framework 2.0。VS 2005 Developer Edition 的典型安装(不安装 Visual J#、Visual C++ 和 Crystal Reports)大约需要 1 GB 的硬盘空间(包括文档)。添加 Visual J#、Visual C++、Crystal Reports 和 SQL Express 时的硬盘空间应为 2.5 GB(无添加的文档)。安装全部文档需要 1 GB 的硬盘空间。用户在一个分区安装 SQL Server 2005 和 VS 2005 Developer Edition 或更高的版本时, 至少应该有 10 GB 的硬盘空间。

运行 VB Express 2005 的最低处理器速度是 600 MHz(建议采用 1 GHz), 最小 RAM 是 128 MB(建议采用 256 MB)。VBX、SQLX、MSDN Online Help 以及 XM 需要约 900 MB 的硬盘空间。示例数据库 Northwind 和 AdventureWorks 需要 125 MB。用户需通过 Microsoft Web 站点下载这两个数据库。用于安装 VBX 和 SQLX 的分区空间至少应是 2 GB。

命名习惯

书中用三字母匈牙利命名法前缀来表示大多数变量的数据类型, 如 `strString`、`intInteger`、`datDateTime` 和 `objObject`。有些两个字母的前缀表示某些类, 如 `cnSqlConnection`、`cmSqlCommand`、`sbStringBuilder`、`xrXmlReader` 和 `xwXmlWriter`。两字母和三字母前缀表示 Windows 和 Web 窗体控件类型, 如 `gbGroupBox`、`btnButton`、`cboComboBox` 以及 `dgvDataGridview`。但自动生成的变量、对象和控件名例外, 如 `DatabaseConnectionString`、`TableTableAdapter` 和 `DatabaseDataSet`。

T-SQL 语句的关键字采用大写, 对象名和变量名采用大小写混合。大多数 XML 元素和属性名采用 `camelCase` 型(不是大小写混合), 以与 SQL Server 表名和列名习惯相对应。

源代码和示例数据库

完成本书各示例时, 可以通过创建所需要的 Windows 窗体或 Windows 页面及其数据源并通过手工添加简要 VB 2005 代码示例的方式来实现较简单的项目。较复杂的示例项目通常取决于本书提供的代码文件中的“启动”项目。本书使用的所有源代码均位于一个容量为 20 MB 的档案文件(VB2005DB.zip)中, 可在 <http://www.wrox.com> 站点下载该文件, 具体方法是: 进入该站点, 找到 One-on-One Visual Basic 2005 Database Programming(利用

Search 文本框或利用某一书名列表来确定), 在详细介绍本书的页面上单击 Download Code 链接, 即可获得本书的全部源代码, 或者访问 www.tupwk.com.cn/downpage 页面获得相关的源代码。

由于许多书有相似的书名, 因此通过 ISBN 进行搜索会更方便些。本书的 ISBN 是 0-7645-7678-X。此外, 用户可以进入 Wrox 的代码下载页(<http://www.wrox.com/dynamic/books/download.aspx>)获取本书以及 Wrox 所有图书的代码。

用于 Northwind 和 AdventureWorks 示例数据库的示例项目的连接字符串将 localhost 指定为 Data Source 或 Server 参数。该参数假设作为用户系统的默认实例安装了 Server 2000/2005 Developer Edition 或更高的版本, 或 MSDE 2000。如果用户没有将 SQL Server 2005 Express 作为默认实例定制安装的话, 需要将 localhost 改为.\SQLEXPRESS, 或将 ServerName\SQLEXPRESS 改为连接到 SQLX。

为示例项目安装源代码

VB2005DB.zip 文件的档案结构由 12 个档案文件组成, 即书的每一章有一个档案文件, 文件名分别为 Chapter01.zip~Chapter12.zip, 它们还包含名称为项目名的子文件夹。ReadMe.txt 文件说明如何对文件解压缩以及如何在示例项目中包含后发布(late-breaking)的信息。有些项目的文件与安装位置有关系, 因此, 每一个 Chapter##.zip 文件为其项目指定了完整的路径名。这些文件被解压缩到%SystemDrive%\VB2005DB\Chapter##\ProjectName 文件夹。如果读者用 WinZip 或类似的档案应用程序解压缩文件, 那么在对文件解压缩之前切记要选中 Use Folder Names 复选框。

安装示例数据库

本书的大多数示例项目都使用了 SQL Server 2000 Northwind 示例数据库, 有些项目需要 SQL Server 2005 版的 AdventureWorks 示例数据库。SQL Server 2005 默认不安装 AdventureWorks OLTP 数据库, 而且 SQLX 不包括示例数据库。下面介绍如何在 SQL Server 2005 Developer Edition 或更高版本的 SQLX 中下载和安装两个示例数据库。

下载示例数据库

编写本书时, 可以对 Northwind、pubs 和 AdventureWorks OLTP 示例数据库进行如下所示的下载链接。

- 在 <http://go.microsoft.com/fwlink/?LinkId=30196> 地址通过 Northwind 和 pubs Sample Databases 页面下载 SQL2000SampleDb.msi, 运行安装程序, 在\Program Files\Microsoft SQL Server 2000 Sample Database Scripts 文件夹中创建 instnwind.sql 和 instpubs.sql T-SQL 脚本。此安装可以向 Programs 菜单添加 Microsoft SQL Server 2000 Sample Database 项。

- 搜索 SQL Server 2005 Express Documentation(包括引号), 从 Microsoft Web 站点下载 AdventureWorksDb.msi。运行安装程序, 创建\Program Files\Microsoft SQL Server 2005 AdventureWorks Sample Database Scripts 文件夹, 并添加 OLTP (awdb)、Analysis Services (awadb)和 data warehouse (awdwdw)文件夹。这些文件夹包含安装每一个 AdventureWorks 版本的 T-SQL 脚本。安装后就会向 Programs 菜单中添加一个 Microsoft SQL Server 2005 AdventureWorks Sample Databases 项。

如果读者通过 SQL Server 2005 Developer Edition 或更高版本的安装程序安装了 AdventureWorks, 则不需要下载它们。

为 SQL Server 2005 Developer Edition 或更高版本安装示例数据库

下面介绍的操作需要先安装 SQL Server Management Studio (SSMS)。如果读者有用于 SQL Server 2005 Express 的 SMSS 的 post-RTM 版本, 则可以用它在 SQLX 中安装示例数据库。如果读者没有 SQLX 图形管理程序, 可跳过“为 SQL Server 2005 Express 安装示例数据库”一节。

用户能够在 Windows XP SP2 或更高的版本中安装 Developer Edition。SQL Server 2005 Standard Edition 和更高的版本需要 Windows 2000 Server 或 Windows Server 2003。用户还能够用 SMSS 的完整版为 SQLX 安装示例数据库。

Northwind 和 pubs

安装 Northwind 数据库及 pubs(可选)的步骤如下。

- (1) 打开 SSMS, 连接到自己的 SQL Server 2005 实例。
- (2) 依次选择 File→Open→File, 导航到\Program Files\Microsoft SQL Server 2000 Sample Database Scripts 文件夹, 双击 instnwind.sql。
- (3) 连接到自己的 SQL Server 2005 实例, 将脚本加载到新查询窗口。
- (4) 单击 Execute 运行项目。
- (5) 重复步骤(2)到步骤(4)(可选), 但要在步骤(2)中选择 instpubs.sql。

本书没有需要 pubs 数据库的示例。

AdventureWorks

AdventureWorks OLTP 数据库的安装与系统驱动器有关。安装此数据库的步骤如下。

- (1) 打开 SSMS, 连接到自己的 SQL Server 2005 实例。
- (2) 关闭 File→Open→File, 导航到\Program Files\Microsoft SQL Server 2005 Adventure Works Sample Database Scripts\awdb 文件夹, 双击 instawdb.sql。
- (3) 连接到自己的 SQL Server 2005 实例, 将脚本加载到新查询窗口。
- (4) 如果用户的系统驱动器不是 C:\, 搜索 “@data_path = 'C:\'” 并用系统驱动器字母替换 C。
- (5) 单击 Execute 运行脚本。

(6) 展开 Object Explorer 的 Databases\AdventureWorks 节点, 右击 Tables 节点之一, 并选择 Open table 验证表已被填充。

如果用户的系统驱动器不是 C:\, 而且用户在步骤(4)中没有编辑脚本, 则会安装数据库, 但不会填充表。

为 SQL Server 2005 Express 安装示例数据库

Paul Flessner, 之后成为 Microsoft 的服务器应用的高级副总, 于 2005 年 12 月 15 日宣布: SQL Server 2005 Express Manager(用于 SQLX 的简化管理应用程序)不再对 SQLS RTM 版有效, 或不再包含于 VS 2005 RTM 的各版本。Paul 承诺: “用于 SQL Server 2005 Express Edition 的 SQL Server 2005 Management Studio 向下伸缩的版本”将在“2006 年上半年推出”。在过渡期, SQLX 用户必须用 SqlCmd 实用工具来运行在自己的 SQLX 实例中安装 Northwind 和 AdventureWorks 的 T-SQL 脚本。当在 SQL Server 2005(默认)和 SQLX 之间更改时, 永久地附加数据库文件(而不是按用户数据库的需求来附加)可简化对示例项目 SqlCommand.ConnectionStrings 的修改。

Northwind 和 pubs

安装 Northwind 数据库和 pubs(可选)的步骤如下。

(1) 打开命令提示, 导航到包含 instnwnd.sql 脚本的文件夹。通常是 C:\SQL Server Sample Databases。

(2) 输入 sqlcmd -S localhost\SQLEXPRESS -i instnwnd.sql, 按 Enter 键执行查询, 该查询会创建 Northwind 数据库, 并添加示例数据。

(3) 输入 sqlcmd -S localhost\SQLEXPRESS, 按 Enter 键输入 SqlCmd 交互模式。

(4) 输入 SELECT * FROM Northwind.dbo.customers, 按 Enter 键, 输入 GO, 并按 Enter 键证实窗口中示例数据库存在。

(5) 重复步骤(2)~(4)(可选), 但在步骤(2)中要用 instpubs.sql 替换 instnwnd.sql, 然后输入 EXIT, 并按 Enter 键返回命令提示。

AdventureWorks

AdventureWorks OLTP 数据库的安装与驱动器有关。如果用户没有将 Microsoft SQL Server 2005 AdventureWorks Sample Database Scripts 文件夹安装在 C:\下, 操作前应将该文件夹复制到 C:\。

安装此数据库及其示例数据的步骤如下。

(1) 打开命令提示, 导航到包含 instawdb.sql 的文件夹, 即 C:\Microsoft SQL Server 2005 AdventureWorks Sample Database Scripts\awdb。

(2) 输入 sqlcmd -S localhost\SQLEXPRESS -i instawdb.sql, 并按 Enter 键执行查询, 该查询会创建 AdventureWorks 数据库, 并添加示例数据。

(3) 输入 sqlcmd -S localhost\SQLEXPRESS, 按 Enter 键输入 SqlCmd 交互模式。

(4) 输入 `SELECT * FROM adventureworks.person.contacttype`, 按 Enter 键, 输入 GO, 按 Enter 键证实窗口中示例数据存在。

(5) 输入 EXIT, 按 Enter 键返回到命令提示。

AdventureWorks 表使用了 SQL Server 2005 的新的用户架构分离功能, 它允许用户用任意前缀代替传统数据库拥有者的名称(默认是 dbo)。AdventureWorks 有 5 个架构: HumanResources、Person、Production、Purchasing 和 Sales。用户必须在 FROM 子句的表名中加上架构前缀。如果忽略了架构, 就会收到错误消息“无效的对象名”。

创建和运行示例项目时需要的硬件

本书的许多示例项目都包括用于报告执行时间的文本框。示例项目的执行时间取决于读者所用计算机的处理器、RAM 有效容量、硬盘性能, 还在一定程度上与操作系统的配置有关。

编写本书时使用的计算机是 Dell PowerEdge 400SC 服务器, 它有 2.261 GHz 的 Pentium 4 的处理器(512 KB 缓存), 1GB 的 RAM, 以及一个 80 GB 的 ATA100 硬盘驱动器。此计算机对下面的操作系统和 SQL Server 实例实现双重开机。

- SQL Server 2005 Developer Edition、SQLX、MSDE 2000 Release A 实例的 Windows Server 2003 Standard Edition。
- SQLX 和 MSDE 2000 Release A 的 Windows XP SP2。

访问网络式 SQL Server 实例和组件的例子通过相对轻流量的开关式 100 Mbps 网络连接到 1GB RAM 的 866 MHz Pentium III。远程服务器运行 Windows Server 2003 Standard Edition。

勘误表

我们已尽最大努力来保证在本书文本或在可下载的项目中没有错误。然而, 人无完人, 错误难免。如果读者发现本书有错误, 如拼写错误或代码错误等, 可将对应信息反馈到邮箱 fwkbook@tup.tsinghua.edu.cn 中, 我们将不胜感激。将勘误表传给我们后, 可以避免其他读者对书中的内容产生误解, 同时还有助于本书质量的提高。

本书的代码示例和示例项目以 VS 2005 Beta 2 和 SQL Server 2005/SQLX 的 post-Beta 2 Community Technical Preview 版本为基础。也许在 RTM 版和本书使用的版本中对这些产品进行了小的修改。示例项目已经被更新到 RTM 版。如果用户用 VS 2005 和 SQL Server 2005 的前 RTM 版本运行项目, 可能会得到意想不到的结果。

p2p.wrox.com

如果想与书的作者和相关人士讨论，可加入位于 p2p.wrox.com 的 P2P 论坛。该论坛是基于 Web 的系统，可使读者发送与 Wrox 公司出版的图书相关的消息和相关的技术，并与其他读者和技术用户交流。当论坛上有一个新帖发布时，它还提供了订阅功能，可通过电子邮件向大家发送大家选择的感兴趣的主题。Wrox 作者、编辑、其他业界专家以及读者均会出现在这些论坛。

在 <http://p2p.wrox.com> 站点，读者还可以发现一些不同的论坛，这些论坛不仅对阅读本书有帮助，而且还对开发自己的应用程序有帮助。加入这些论坛的步骤如下。

- (1) 进入 p2p.wrox.com 站点，单击 Register 链接。
- (2) 阅读使用项并单击 Agree。
- (3) 完成加入论坛所需要的信息以及希望提供的可选信息，并单击 Submit。
- (4) 接收一个提供如何验证账户以及如何完成加入过程方面的信息的 E-mail。

不需要加入 P2P 就能够阅读这些信息。但如果用户需要发送自己的消息，则必须加入 P2P。

一旦加入论坛，就能够发送消息并响应其他用户发送的消息。用户能够在任意时间通过 Web 阅读消息。如果用户希望利用电子邮件将专题论坛中的新消息发送给您，在论坛列表中单击论坛名旁的 Subscribe to this Forum 图标即可。

阅读 P2P FAQ 可得到如何使用 Wrox P2P 方面的更多信息。通过 P2P FAQ，可以得到论坛软件以及 P2P 和 Wrox 图书方面的许多公共问题的解答。在任一 P2P 页面上单击 FAQ 链接，即可阅读 FAQ。

目 录

第 I 部分 ADO.NET 2.0

基础知识

第 1 章 从 ADO 向 ADO.NET 移植 3

1.1 数据访问的新方式 4

1.1.1 System.Data 命名空间 4

1.1.2 ADO.NET 数据提供者 6

1.2 ADO.NET 的基本数据对象 6

1.2.1 用 SqlConnection 创建基本 ADO.NET 数据对象 8

1.2.2 对多表更新应用事务 15

1.2.3 使用 OleDb、SqlXml 和 Odbc 成员类 17

1.2.4 使用类型化 DataReader 数据 和 SqlResultSet 数据 21

1.3 ADO.NET 的类型化 DataSet 对象 22

1.3.1 通过 SQL Server 数据源添加 类型化 DataSet 24

1.3.2 添加 DataGridView 和 BindingNavigator 控件 28

1.3.3 持久化和重新打开 DataSet 30

1.3.4 将 DataViewGrid 更改为 Details 窗体 31

1.3.5 添加相关的 DataBound 控件 32

1.4 小结 34

第 2 章 ADO.NET 2.0 新功能介绍 35

2.1 使用新的 ADO.NET 2.0 运行 时 Windows 窗体对象 35

2.1.1 用 DbProviderFactories 创建 数据库不明确的项目 36

2.1.2 检索基表架构 39

2.1.3 校验有效的 SQL Server 实 例和 ADO.NET 2.0 数据 提供者 41

2.1.4 利用 SqlBulkCopy 对象对 SQL Server 表进行批量 插入 42

2.1.5 SQL Server 连接统计信息 45

2.1.6 异步执行 SqlCommands 47

2.1.7 创建单独的 DataTable 56

2.1.8 用 Nullable 类型支持 DBNull 值 60

2.2 使用 ADO.NET 2.0 的新持久 性 Windows 窗体对象 64

2.2.1 ADO.NET 1.x 与 2.0 的数据 设计器比较 64

2.2.2 向工具箱添加没显示的 ADO.NET 控件 66

2.2.3 将 1.x 项目升级为 ADO.NET 2.0 组件 67

2.2.4 添加多级子窗体 67

2.2.5 参数化 MasterDetailsForm 69

2.2.6 批量多次更新 71

2.3 用 ReportViewer 控件设计和 显示报表 72

2.4 小结 74

第 3 章 对以数据为中心的项目采用 最佳实践 77

3.1 建立体系结构的最佳实践 78

3.2 为项目确定模式 80

3.2.1 使用 Microsoft .NET 的企业 解决方案模式 81

3.2.2 数据模式 81

3.2.3	分布式系统模式	82
3.2.4	集成模式	82
3.3	应用程序块库	83
3.3.1	数据访问应用程序块	84
3.3.2	DataAccessQuickStart 客户端	88
3.4	遵循设计指南	90
3.4.1	.NET 数据访问体系结构 指南	90
3.4.2	提高.NET 应用程序的性能 和可伸缩性	91
3.4.3	设计数据层组件并在层间 传递数据	91
3.5	类库设计指南	96
3.5.1	命名指南	96
3.5.2	类成员使用指南	97
3.6	面向服务的架构	98
3.6.1	面向服务的架构	98
3.6.2	用 Web 服务实现 SOA	99
3.6.3	保证完全互操作的 Web 服务	100
3.7	用 FxCop 验证项目代码	104
3.8	自动完成测试驱动的开发	106
3.8.1	向 VS 2005 解决方案添加 单元测试项目	106
3.8.2	编辑和运行向导生成的 测试	108
3.9	运行 SQL Server 2000 最佳 实践分析器	109
3.10	对 ADO.NET 2.0 项目采用 特殊最佳实践	111
3.10.1	对存储池数据库连接采用 相同连接字符串	111
3.10.2	运行 SQL Server 分析器来 检验 SQL 和 RPC 查询	112
3.10.3	避免添加运行时 CommandBuilder 实例	113
3.10.4	用存储过程代替 SQL 批量 查询	114

3.10.5	为不需要的参数添加 默认值	114
3.10.6	利用 sp_executesql 和命名 参数重新使用缓存的查询 计划	114
3.10.7	向开放式并发控件添加 timestamp 列	115
3.10.8	在并发测试中检验所有 关联记录	118
3.10.9	通过批量更新减少服务器 往返行程	118
3.10.10	利用客户端验证避免 SqlExceptions	118
3.11	小结	119

第 II 部分 Windows 窗体和控 件中的数据绑定

第 4 章	对 TableAdapters、Binding Sources 和 DataGridView 编程	123
4.1	设计基本的客户-订单-订单 详细窗体	123
4.1.1	利用参数化查询减小 DataSet 的尺寸	124
4.1.2	创建数据源并添加控件	125
4.1.3	为各数据表添加 FillBy 方法	126
4.1.4	更改自动生成的代码来满足 控件的要求	128
4.1.5	用 CustomerID 值填充 组合框	129
4.1.6	清除 UI 和代码	130
4.2	格式化并添加 DataGridView View 列	131
4.2.1	设置 OrdersDataGridView 列的格式	131
4.2.2	格式化 Order_DetailsDataGridView 并添加计算列	132

4.3 为新记录提供默认值	134	4.11.2 创建并测试 UpdateBase Tables 函数	160
4.3.1 添加默认的订单记录值	134	4.12 小结	168
4.3.2 添加默认的详细记录值	135	第 5 章 添加数据验证和并发管理	171
4.4 处理 DataErrors 事件	136	5.1 验证数据输入	173
4.5 流线型 Heads-down 数据 输入	137	5.1.1 验证文本框	173
4.6 将 UI 移植到选项卡式窗 体上	139	5.1.2 验证 DataGridViews	174
4.6.1 测试 OrdersByCustomersV3 项目	140	5.1.3 发现输入中的主键约束 冲突	176
4.6.2 用代码添加行时修改丢失的 默认值	141	5.1.4 验证默认值	177
4.6.3 在第二个选项卡页面编辑选 择的 DataGridView 记录	142	5.2 管理并发冲突	178
4.7 为主键值创建和绑定查询 列表	143	5.2.1 ADO.NET 2.0 并发控制和 事务变更	179
4.7.1 创建非类型化查询 DataSet 及其 DataTables	144	5.2.2 并发控制战略	181
4.7.2 填充 cboCustomerID 组合框	146	5.2.3 并发管理的“链接丢失”	181
4.7.3 用组合框代替 DataGridView 文本框	146	5.2.4 预测基于值的主键约束 冲突	189
4.7.4 为组合框与文本框建立 关联	149	5.3 合理处理并发错误	191
4.8 添加设置附加值的组合框	151	5.3.1 从服务器上获取当前数据	192
4.8.1 创建和绑定根据 ProductName 排序的 DataView	151	5.3.2 检索并比较服务器和客户端 单元值	194
4.8.2 测试复制性并更新 UnitPrice 列	152	5.4 容纳非连接用户	198
4.9 为新客户输入添加查询表行	155	5.4.1 创建和管理脱机 DataSet	198
4.9.1 添加和绑定 CustomerID BindingSource	155	5.4.2 处理多个父记录	200
4.9.2 用 DataRowView 测试 重复性	156	5.5 小结	201
4.10 对编辑应用业务规则	157	第 6 章 高级 DataSet 技术应用	203
4.11 将更改保存到基表	159	6.1 对 DataSet 更新应用事务	203
4.11.1 维护引用的完整性	159	6.2 对 DataTable 选择命令添加 联接	211
		6.2.1 向 SelectCommand 添加 联接	211
		6.2.2 向 DataGridView 添加联接 的列	213
		6.2.3 提供默认值并更新只读列	213
		6.3 通过减少 DataSet 的尺寸来 提高性能	215
		6.3.1 由 TOP n 查询返回的 限制行	216