

新世纪电脑丛书

Visual C++数据库开发技巧与实例

冯华 编著

提供 Visual C++ 数据库开发的完整方案

通过实例讲解 VC 数据库的开发步骤

根据实际经验介绍 Visual C++ 的应用技巧

书中实例均由作者精选或亲自制作而成

随书赠送的光盘资源丰富，即插即用

新世纪电脑丛书

Visual C++ 数据库开发技巧与实例

冯华 编著



机械工业出版社

本书是一本介绍如何利用 Visual C++ 开发数据库应用的专业书籍，它融入了作者从事 Visual C++ 数据库开发的经验与技巧，通过大量的实例介绍了 Visual C++ 开发数据库应用的主要技术方法和解决方案，包括 ODBC、DAO 以及 OLE DB 等数据库的应用技术。

本书共分 10 章，重点介绍了如何利用 Visual C++ 通过 ODBC、DAO 以及 OLE DB 等技术方法开发数据库应用。本书在对这些数据库技术进行深入剖析的基础上，通过实例翔实地介绍了 Visual C++ 开发数据库应用的思路、步骤和程序框架，循序渐进地引导读者了解和掌握 Visual C++ 的数据库开发技术，并介绍了 Visual C++ 的数据库工具 Database Project。另外，书中还介绍了数据库的一些相关知识以及数据库应用项目的规划与管理，较为详细地介绍了数据库系统 SQL Server 7.0，包括 SQL Server 7.0 数据库以及数据库对象的管理、用户管理等。

本书力求从全面、实用和易读的角度出发，结合实例介绍 Visual C++ 的数据库开发技术，同时兼具知识性、技术性和实用性，为广大读者提供借鉴。本书可供对 Visual C++ 有所了解，并从事数据库开发工作的技术人员以及在校的本科生、研究生使用。

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）

责任编辑：何月秋 封面设计：艾迪 艾藤

责任印制：何全君

三河市宏达印刷厂印刷·新华书店北京发行所发行

2000 年 7 月第 1 版第 1 次印刷

787mm×1092mm 1/16 · 20.25 印张 · 493 千字

0 001— 5000 册

定价： 37.00 元 （ICD，含配套书）

ISBN 7-900043-28-4/TP·28

Email: sbs@mail.machineinfo.gov.cn

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

本社购书热线电话（010）68993821、68326677-2527

前言

近几年来,随着计算机技术的发展,涌现出了各种数据库编程语言以及开发方法,它们为数据库的开发提供了更为广泛的选择空间。在众多的编程语言中,Microsoft 推出的 Visual C++应该是最闪亮的明星之一。Visual C++依据崭新的编程思路、强大的功能,以及包罗万象的类库,备受编程人员的青睐,在许多开发领域得到了广泛的应用。Visual C++中提供了多种方便易用的数据库开发途径和功能强大的类库,增加了数据库开发的灵活性,减轻了开发的复杂性,因此,在很多数据库开发工作中,Visual C++发挥着重大的作用。

本书从全面、实用和易读的角度出发,结合作者利用 Visual C++开发数据库应用的经验,通过实例的形式介绍了 Visual C++开发数据库应用的主要技术方法和解决方案。其中,详细介绍了利用 Visual C++通过 ODBC、DAO 以及 OLE DB 技术方法开发数据库的应用。ODBC、DAO 以及 OLE DB 是近年来最为常用的数据库开发技术,Visual C++对它们提供了全面的支持,在 Visual C++的类库中定义了丰富的 ODBC、DAO 和 OLE DB 类,并封装了大量的成员变量和函数,它们能够完成大部分数据库操作。本书在对使用 Visual C++通过 ODBC、DAO 以及 OLE DB 技术开发数据库应用进行深入剖析的基础上,给出了相应完整的应用实例。这些实例提供了从 Visual C++数据库应用的创建到完成各种数据库操作功能的完整的解决方案,具有很高的通用性和实用价值。

另外,本书还针对已有的不具备数据库功能的 Visual C++应用程序,介绍了如何在其现有应用的基础上添加数据库功能,使其能够访问特定的数据库。

数据库应用离不开数据库系统,要开发出好的数据库应用必须要了解和掌握所使用的数据库系统。因此,本书还结合应用实例中所使用的数据库系统,有针对性地介绍了数据库以及开发的相关知识,如数据库构造、SQL 语句、开发项目的规划与管理等,并介绍了如何创建应用实例中所使用的数据库,包括 Access 数据库和 SQL Server 7.0 数据库,特别详细地介绍了数据库 SQL Server 7.0,包括数据库以及数据库对象的管理、用户管理等。本书的最后还介绍了 Visual C++的数据库工具 Database Project,它可以直接连接并访问数据库对象,是 Visual C++数据库开发的一个很好的辅助工具。

总之,本书结合作者从事 Visual C++数据库开发的经验,通过实例翔实地介绍了 Visual C++开发数据库应用的思路、步骤和程序框架,循序渐进地引导读者了解和掌握 Visual C++的数据库开发技术,解决开发过程中遇到的实际问题,兼具知识性、技术性和实用性,希望能给读者提供借鉴。由于本书作者水平的局限性,书中难免存在一些问题,希望读者批评指正,共同探讨。

编 者

目 录

前言

第 1 章 Visual C++与数据库的连接	1
1.1 ODBC	1
1.1.1 ODBC 概述	1
1.1.2 ODBC 的工作原理	2
1.1.3 ODBC 的结构	3
1.1.4 ODBC 驱动程序	3
1.1.5 16 位和 32 位 ODBC 驱动程序	6
1.1.6 ODBC 数据源	7
1.2 DAO	13
1.3 DAO 与 ODBC 的比较	14
第 2 章 Visual C++开发 ODBC 数据库应用	15
2.1 应用实例概述	15
2.2 Visual C++中的 ODBC 类	16
2.2.1 CDatabase	16
2.2.2 CRecordset	17
2.2.3 CRecordView	19
2.3 使用 AppWizard 创建 ODBC 应用程序	20
2.3.1 建立数据源	20
2.3.2 使用 AppWizard	24
2.3.3 创建视图	31
2.3.4 应用程序如何工作	35
2.4 使用 ClassWizard 创建 CRecordset 和 CRecordView 类	39
2.4.1 创建 CRecordset 类	39
2.4.2 创建 CRecordView 类	44
2.4.3 显示 CCustView	47
2.5 数据的遍历、查找和排序	51
2.5.1 数据的遍历	51
2.5.2 数据的查找	54
2.5.3 数据的排序	59
2.6 数据的操作	60
2.6.1 数据的添加与更新	60
2.6.2 数据的删除	65
2.7 多表操作	68

2.7.1 多表连接.....	69
2.7.2 创建视图.....	71
2.7.3 创建函数.....	74
2.7.4 运行程序.....	79
2.7.5 参数化查询	80
2.8 增加统计功能.....	83
2.9 动态连接数据库表.....	88
2.10 连接多个数据源.....	95
2.11 异常处理.....	99
第 3 章 Visual C++ 开发 DAO 数据库应用.....	101
3.1 应用实例概述.....	101
3.2 DAO 类	102
3.2.1 CDaoWorkspace.....	102
3.2.2 CDaoDatabase	103
3.2.3 CDaoRecordset	104
3.2.4 CDaoRecordView.....	105
3.3 使用 Wizard 建立 DAO 数据库应用程序	106
3.3.1 建立连接.....	106
3.3.2 使用 AppWizard 创建应用	106
3.4 创建视图和数据操作.....	115
3.4.1 创建视图.....	115
3.4.2 数据操作.....	117
3.5 创建一个网格视图.....	126
3.5.1 添加 Microsoft FlexGrid Control 控件.....	126
3.5.2 创建带有网格的视图.....	127
3.5.3 在网格视图中显示数据.....	128
3.5.4 显示网格视图	137
3.6 多表操作.....	141
3.6.1 修改带有网格的视图.....	141
3.6.2 处理多表操作.....	142
3.7 报表打印.....	157
第 4 章 给已有的程序添加数据库应用.....	162
4.1 Visual C++ 的 ODBC 和 DAO 应用剖析.....	162
4.2 应用实例概述.....	165
4.3 已有的 Visual C++ 应用通过 ODBC 访问数据库.....	167
4.4 已有的 Visual C++ 应用通过 DAO 访问数据库.....	176
4.5 使用 DBGrid 控件.....	183

4.5.1 插入控件 DBGrid 和 RemoteData	184
4.5.2 创建对话框	186
4.5.3 设置控件属性	187
第 5 章 Visual C++ 中 OLE DB 的应用	194
5.1 OLE DB 的概念	194
5.1.1 什么是 OLE DB	194
5.1.2 OLE DB 和 ODBC 之间的关系	195
5.2 应用实例概述	195
5.3 Visual C++ 中的 OLE DB 类	196
5.3.1 CDataSource	196
5.3.2 CSession	197
5.3.3 CRowSet	197
5.3.4 CTable	199
5.4 Visual C++ 的 OLE DB 应用实例	200
5.4.1 创建应用程序	200
5.4.2 显示数据库表	204
5.4.3 显示表定义	213
5.4.4 显示表记录	223
第 6 章 关系数据库概述	232
6.1 关系模型的结构	232
6.1.1 数据结构	232
6.1.2 完整性规则	232
6.1.3 数据操作	233
6.2 关系数据库的构造元素	235
6.2.1 表	235
6.2.2 视图	235
6.2.3 索引	236
6.3 创建一个 Access 数据库实例	237
6.3.1 创建数据库	237
6.3.2 创建表	238
6.3.3 输入数据	243
6.3.4 创建关系	243
6.3.5 创建窗体	245
第 7 章 结构化查询语言 — SQL	249
7.1 数据定义语言	249
7.1.1 Create	249

7.1.2	Alter	250
7.1.3	Drop	251
7.2	数据控制语言	251
7.2.1	Grant	251
7.2.2	Revoke	252
7.3	数据操作语言	252
7.3.1	Select	252
7.3.2	Insert	256
7.3.3	Delete	257
7.3.4	Update	257
第 8 章	Visual C++数据库项目规划与设计	259
8.1	项目规划	259
8.1.1	收集项目信息	259
8.1.2	组建项目组	260
8.1.3	划分项目阶段	261
8.2	设计应用	261
8.2.1	需求分析	261
8.2.2	选择体系结构	262
8.2.3	确定数据库	263
8.2.4	定义类以及相关函数和属性	263
8.2.5	建立用户界面	263
8.2.6	建立编程约定标准	264
第 9 章	SQL Server 7.0 的管理与应用	265
9.1	SQL Server 7.0 的概述	265
9.2	SQL Server 7.0 的主要组件	267
9.2.1	Enterprise Manager	267
9.2.2	Service Manager	268
9.2.3	Profiler	269
9.2.4	Query Analyzer	269
9.2.5	Client and Server Network Utility	270
9.3	数据库的使用与管理	271
9.3.1	使用文件和文件组	271
9.3.2	系统数据库	272
9.3.3	创建数据库	273
9.3.4	修改数据库	278
9.3.5	删除数据库	280
9.4	数据库对象的使用与管理	

9.4.1 数据库表.....	281
9.4.2 视图.....	287
9.4.3 索引.....	290
9.4.4 存储过程.....	292
9.4.5 触发器	295
9.5 SQL Server 7.0 的用户管理	296
9.5.1 SQL Server 7.0 的用户安全机制	296
9.5.2 用户管理	298
第 10 章 Visual Database Tools.....	305
10.1 Visual Database Tools 概述.....	305
10.2 创建 Database Project	305
10.3 操作数据库对象.....	310
10.3.1 表和视图的操作	310
10.3.2 存储过程	315

第1章 Visual C++与数据库的连接

Visual C++作为一种功能强大的应用软件开发平台，它在数据库应用中的开发也是首屈一指的。本章将主要介绍目前利用 Visual C++开发数据库应用的主要方法——ODBC 和 DAO，简要介绍 ODBC 和 DAO 的一些概念，使读者能够对如何利用 Visual C++开发数据库应用有一个全面认识。

1.1 ODBC

开放数据库互联(Open Data Base Connectivity, 简称 ODBC)是一个用于访问数据库的统一界面标准。为了使读者能够更好地在 Visual C++中利用 ODBC 方法开发数据库应用，本节将主要介绍 ODBC 的基本概念、原理和结构，以及如何建立 ODBC 数据源内容。相信读者通过对本节的学习，能够更好地认识 ODBC 及其应用方法。

1.1.1 ODBC 概述

开放数据库互联(Open DataBase Connectivity, 简称 ODBC)实际上是一个数据库访问库，可以使应用程序直接操纵数据库中的数据，具有数据库的独立性。ODBC 是微软公司 WOSA(Windows Open Service Architecture)的组成部分，是微软公司提出的开放式数据库互联的标准接口，用户可以通过加载连接到的数据库的驱动程序来建立与各种数据库的连接。使用 ODBC 不仅可以访问 Access、SQL Server、Oracle、Sybase、Lotus Notes 等数据库，而且可以访问 Excel 电子表格以及 ASCII 数据文件等非数据库对象，这也正是 ODBC 的独特之处。

ODBC 通过使用驱动程序(Driver)提供了很好的数据库独立性，驱动程序与具体的数据库有关。这样，通过 ODBC 开发的数据库应用程序，如果想更换所使用的数据库，移植到其他数据库平台上是非常容易的，例如以前的应用程序使用的是 Access 数据库，现在希望将其移植到 SQL Server 数据库上，只需将应用程序改换一下驱动程序即可，也就是说应用程序不再使用 Access 数据库的 ODBC 驱动程序，而使用 SQL Server 数据库的 ODBC 驱动程序即可移植到 SQL Server 数据库平台上。由此可见，使用 ODBC 可以带来诸多益处：

- ODBC 的数据库独立性使得应用程序移植到其他数据库上变得非常容易，避免了应用程序随数据库改变而改变的烦恼。
- 使用 ODBC 可以缩短应用程序的开发时间。例如现在需要开发一个基于 Client/Server 结构的大型数据库应用系统，如果使用 ODBC，我们不必把开发人员集中到其环境下工作，可以先在单机版数据库(如 SQL AnyWhere)上开发，然后再更改到真正的运行环境和数据库系统上，这样就给开发工作提供了更加灵活的方式，加快了开发进程。
- 使用 ODBC 可以避免应用程序对不同类型的数据库使用不同的 API。不同类型的数据库具有不同的 API 库，它们都大同小异，但学习起来比较浪费时间，且容易混淆，而使用 ODBC 的应用程序只用 ODBC 的 API，一般不需要数据库本身的 API，除非当用 ODBC 编写的程序影响了系统的性能时，要考虑用数据库本身的 API 以获得更好的性能。

1.1.2 ODBC 的工作原理

在 1.1.1 节中提到 ODBC 是通过使用驱动程序(Driver)提供数据库独立性的。驱动程序是 ODBC 的一个重要组件，它是一个用以支持 ODBC 函数调用的模块，通常是一个 DLL，并与具体的数据库有关，例如操作 Access 数据库要使用 Access 的 ODBC 驱动程序，而要操作 SQL Server 数据库必须使用 SQL Server 的 ODBC 驱动程序。应用程序是通过调用驱动程序所支持的函数来操作数据库的，因此，如果应用程序要操作不同类型的数据库，就要动态地连接到不同的驱动程序上。

ODBC 的另一个重要组件是驱动程序管理器(Driver manager)，它负责管理 ODBC 应用程序中 ODBC 函数与 DLL 中函数之间的绑定(Binding)，驱动程序管理器包含在 ODBC.DLL 或 ODBC32.DLL 中，可连接到所有的 ODBC 应用程序中。

许多应用程序开发软件都能提供 ODBC 接口，Visual C++ 就是其中的一种。ODBC 接口是以 SQL 作为标准的查询语言来存取连接到的数据源，它允许单个的应用同时访问不同的数据库管理系统(DBMS)，这使应用程序的开发人员可以不必关注所操作的数据库管理系统究竟是什么。图 1-1 给出了 Visual C++ 通过 ODBC 接口连接到各种数据库管理系统的调用层次。

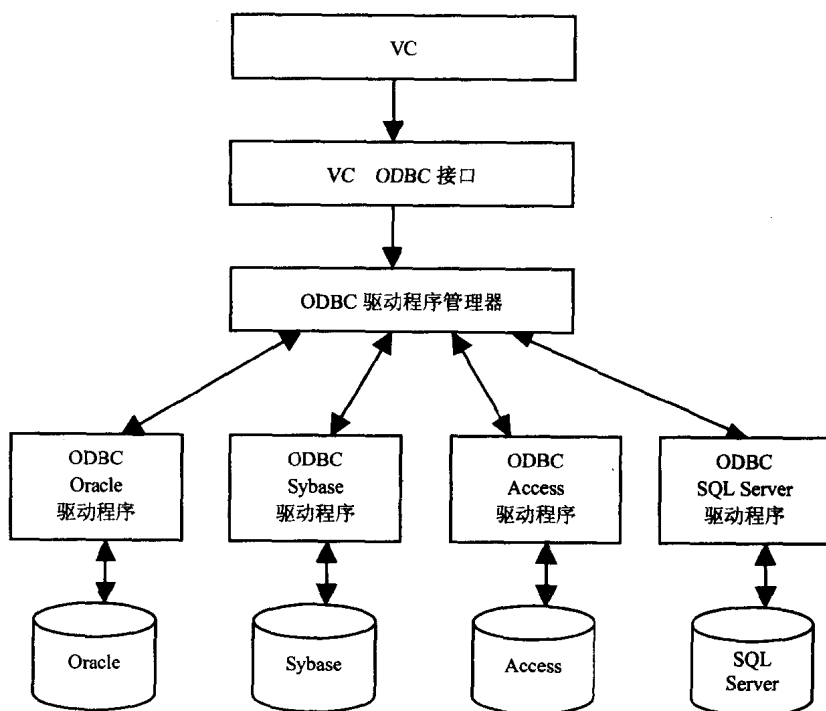


图 1-1 Visual C++ 通过 ODBC 接口连接到各种数据库管理系统的调用层次

Visual C++ 2.0 及其以上版本中包含了 ODBC 库和头文件，并在 Microsoft 基本库，即 MFC 类库中包含了基于 ODBC 的扩展数据库类，它封装了使用 ODBC 资源的细节，替开发

人员完成了大部分工作，使 ODBC 应用程序的开发变得容易得多，我们将在后面的章节中详细介绍如何通过 Visual C++ 的 Wizard 和 MFC 类库开发 ODBC 应用。

1.1.3 ODBC 的结构

由 1.1.2 节介绍的 ODBC 工作原理，我们可以看到 ODBC 调用主要由四个层次组成，如图 1-2 所示。

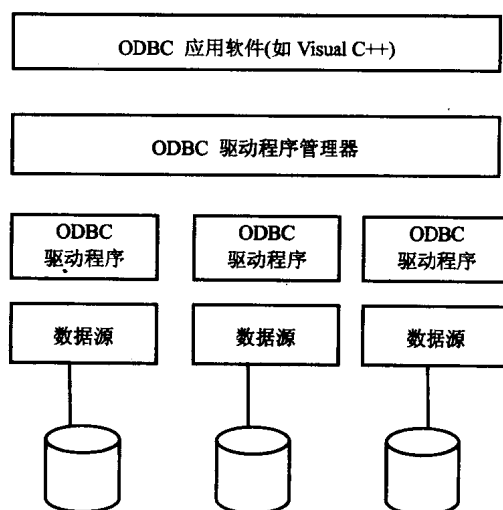


图 1-2 ODBC 四层结构

(1) 第一层：ODBC 应用软件(如 Visual C++) 它通过调用 ODBC 函数向数据库发送 SQL 语句并处理 SQL 返回结果。

(2) 第二层：ODBC 驱动管理程序 它负责管理和装载驱动程序，其作用是：

- 用 ODBC.INI 文件映射数据源名到特定的驱动程序。
- 处理 ODBC 的初始化调用。
- 为每个驱动程序提供 ODBC 函数的入口点。
- 对 ODBC 的调用参数和顺序进行检查。

(3) 第三层：ODBC 驱动程序 处理 ODBC 函数调用，提交 SQL 请求给特定的数据源并返回结果给应用。如果必要，它将修改 SQL 请求，使得 SQL 请求的语法与特定数据库的语法一致。

(4) 第四层：数据源 数据源是指要存取的数据及其相关的操作系统、数据库管理系统和网络系统。

1.1.4 ODBC 驱动程序

在 ODBC 应用程序中不能直接存取数据库中的数据，必须通过 ODBC 驱动程序管理器将应用程序的 SQL 语句及其他信息传递给 ODBC 驱动程序，再由驱动程序负责将结果集返回给应用程序。

ODBC 驱动程序作为应用程序和特定数据库之间的接口，它是一种动态连接库(DLL)，在这一点上不同于通常的数据库，但在其他方面它们没有什么区别。对于同一个驱动程序支持的数据库，ODBC 都提供基本统一的接口。根据功能和运行模式，驱动程序可以分为很多种，一般来说，可以按 API 一致性级别、SQL 语句一致性和驱动程序类型分类。

1. API 一致性

API 一致性级别决定了应用程序所能调用的函数种类。驱动程序不可能实现所有的 ODBC 函数，一般可达到以下三个级别：

- API 一致性的核心级；
- API 一致性的扩展级 1；
- API 一致性的扩展级 2。

这三个级别只是一种指导性原则，某些驱动程序实际上并没有到达它们所到达级别的所有函数。核心级的 API 一致性仅包括最基本的 23 个函数，这些函数构成了驱动程序的核心，包括分配和释放环境句柄、数据库连接、执行 SQL 语句等。此外，核心级的 API 一致性还提供了向语句中传入参数、存取执行结果等一些基本功能，核心级的 API 一致性驱动程序仅能满足最基本的应用程序需求。

API 一致性的扩展级 1 是在核心级的基础上增加了 19 个函数，可以在应用程序中动态地了解表的模式。

API 一致性的扩展级 2 是在扩展级 1 的基础上又增加了 19 个函数，通过这些函数可以了解到关于数据库主码和外码的信息、表和列的权限信息以及存储过程信息，另外，扩展级 2 具有更强的游标和并发控制功能。

现在，大多数驱动程序都已支持到扩展级 1，而在扩展级 2 的许多功能仅适用于 Client/Server 结构的数据库，只有应用程序运行在真正的 Client/Server 结构的数据库平台上，扩展级 2 才能发挥实际作用。

2. SQL 语句一致性

驱动程序的 SQL 语句一致性级别决定了 ODBC 语句中所能使用的 SQL 语法的类型和数据类型。SQL 一致性可分为以下三个级别：

- 基本 SQL 语法；
- 核心 SQL 语法；
- 扩展 SQL 语法。

基本 SQL 语法可以满足应用程序的大多数需求，即可在应用程序中执行创建(Create)和撤消(Drop)表，选择(Select)、更新(Update)和删除(Delete)记录等操作，并支持许多字符型数据类型，如 char、varchar、longvarchar 等。

核心 SQL 语法在基本 SQL 语法的基础上增加了很多功能，可以修改(Alter)表、创建和撤消(Drop)视图、授权(Grant)和撤消(Revoke)给用户的各种权限。此外，核心 SQL 语法支持嵌套子查询，并增加了整型、浮点型等数据类型。

扩展 SQL 语法引入了一些较复杂的语法，如游标控制等。一般来说，一个驱动程序必须要能支持核心 SQL 语法的大部分内容。

3. 驱动程序类型

驱动程序类型决定了驱动程序与相关的数据库系统的分工程度，可分为单束式和多束式

驱动程序。

(1) 单束式驱动程序 单束式驱动程序类似于本地的数据库管理系统，它介于应用程序和数据库之间，充当数据库的引擎，为非 Client/Server 结构的数据提供一个统一的通信方式，它既处理 ODBC 调用，又处理 SQL 语法。图 1-3 描述了单束式驱动程序的功能。

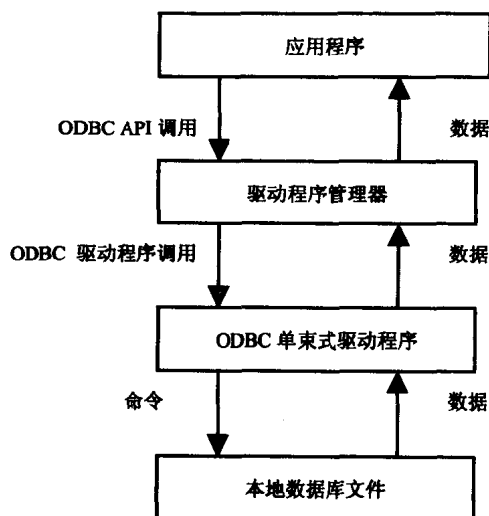


图 1-3 单束式驱动程序的功能

当应用程序要操作数据库时，首先传递一个 ODBC 函数调用给 ODBC 驱动程序管理器，驱动程序管理器负责管理应用程序和 ODBC 驱动程序之间的通信，并决定下一步要采取的操作，如果驱动程序管理器能够直接处理此函数调用，那么它将在处理完毕后把结果直接送给应用程序；如果驱动程序管理器需要调用 ODBC 驱动程序中的函数，则执行调用命令，并将结果返回给应用程序。

从以上可以看到，单束式驱动程序实际上起到的是数据库引擎的作用，它不需要其他的帮助即可完成相应的工作。dBASE 驱动程序就是一个典型的单束式驱动程序，它不仅能像独立的数据库管理系统一样操作 dBASE 的表和索引，而且能对 SQL 语句进行分析，并转化为相应的函数调用，可以说单束式驱动程序实际上就是数据库管理系统。

(2) 多束式驱动程序 多束式驱动程序本身并不执行数据处理操作，而是负责在数据库引擎和客户应用程序之间传递命令和数据。图 1-4 描述了多束式驱动程序的功能和作用。

当应用程序对数据库发出操作请求时，首先传递给驱动程序管理器，驱动程序管理器根据操作请求，或自己处理请求，或提交给驱动程序，如果它把请求提交给驱动程序，则由多束式驱动程序将请求翻译成指定的数据库通信接口(SQL Server's DB_Library)所接受的形式，并交给接口处理，接口通过网络把请求发送给服务器上的数据库引擎，服务器处理完请求后把结果返回给数据库通信接口，再由数据库接口将结果传给多束式驱动程序，并由多束式驱动程序将结果传递给应用程序。

由以上我们可以看到，多束式驱动程序与数据库管理系统是分离的，它只负责处理应

用程序中 ODBC 函数调用，而具体的 SQL 语句处理则由数据库管理系统去完成。多束式驱动程序通常还可充当具有 Client/Server 结构的数据库管理系统的通信管理器，它并不直接操作数据文件，而是通过本地数据库系统的客户端与服务进行通信，驱动程序对 SQL 语句不做任何处理，而只是把 SQL 语句直接传递给数据库系统的客户端。

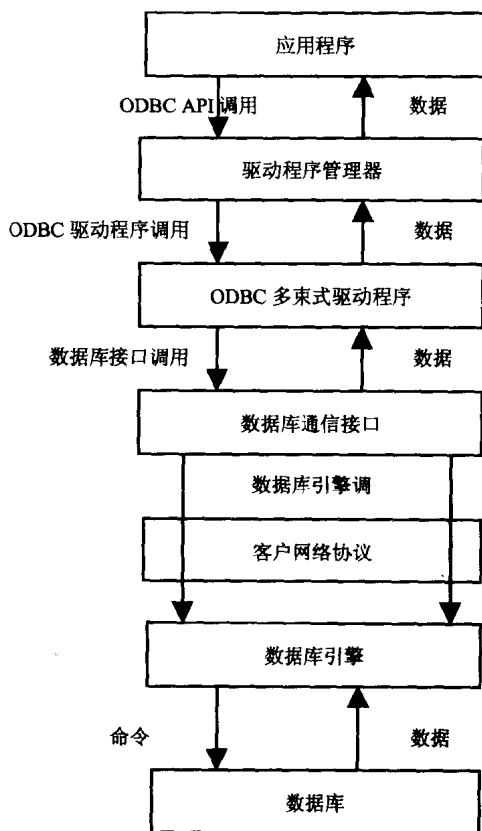


图 1-4 多束式驱动程序的功能和作用

1.1.5 16 位和 32 位 ODBC 驱动程序

ODBC 驱动程序有 16 位和 32 位驱动程序之分。最初 ODBC 工作在 Windows 平台上，Windows 是 16 位的，因此 ODBC 驱动程序是 16 位的。但随着 Windows NT、Windows 95/98 的相继推出，也出现了 32 位的 ODBC 驱动程序。16 位和 32 位驱动程序在速度、性能以及多线程等方面有许多差别。

- 速度：32 位 ODBC 驱动程序要比 16 位 ODBC 驱动程序快。
- 性能：32 位驱动程序提供了一些 16 位驱动程序不具备的性能，如大多数 32 位驱动程序允许同一时刻管理多个任务。
- 多线程：多线程是 32 位平台的一个重要特性，32 位驱动程序可以充分利用这一特性，例如，可以同时指定多个线程操作数据库中的数据，主线程可以重新初始化发生错误的

进程，直至进程成功或要求其终止。

1.1.6 ODBC 数据源

开发 ODBC 应用程序首先要做的就是建立 ODBC 数据源，它告诉 ODBC 应用程序将操作哪种数据库或数据库文件以及所使用的 ODBC 驱动程序。在目前使用最为广泛的 Windows 95/98 操作系统下，打开控制面板，将会看到一个 ODBC 图标，双击该图标打开“ODBC 数据源管理器”对话框，如图 1-5 所示。

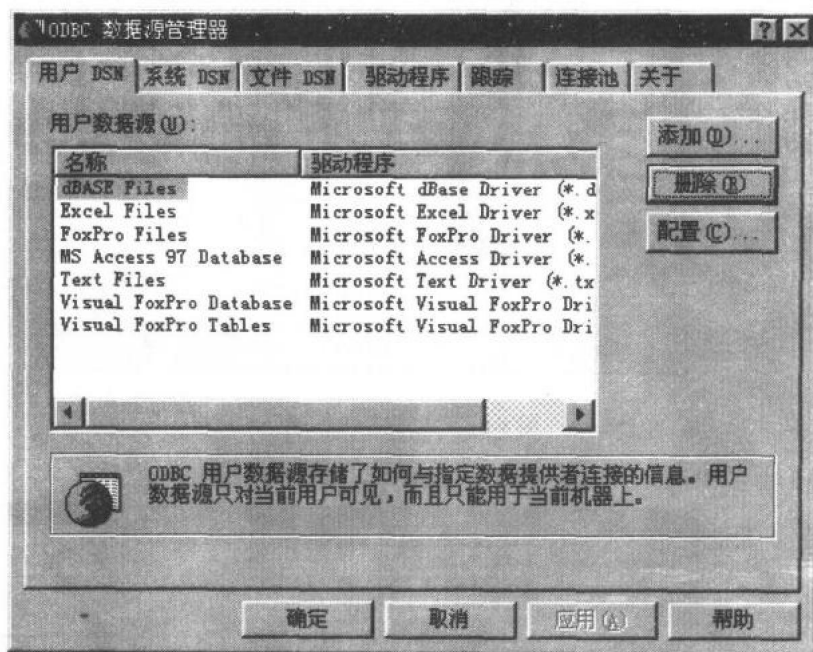


图 1-5 “ODBC 数据源管理器”对话框

在这里能够查看已安装的 ODBC 驱动程序，并可以建立数据源。下面就介绍一下 ODBC 数据源管理器中的属性页以及如何建立数据源。

1. 用户 DSN

在“用户 DSN”页中列出了已建立的用户数据源，在这里可以创建新数据源，也可以对已有的数据源进行修改和删除。创建一个新数据源的步骤如下：

(1) 单击“添加”按钮，出现如图 1-6 所示的“创建新数据源”对话框，列出了所有已安装的 ODBC 驱动程序，从中选择所需要的驱动程序，例如选择 Microsoft Access Driver，并单击“完成”按钮，出现 ODBC Microsoft Access 97 Setup 对话框。

(2) 在 ODBC Microsoft Access 97 Setup 对话框的 Data Source Name 字段中输入数据源的名字，如 sale，在 Description 字段中可以简要描述一下数据源，也可以为空。单击 Select 按钮选择一个已存在的 Access 数据库，如 D:\DBSample\sale.mdb，如图 1-7 所示。在 ODBC Microsoft Access 97 Setup 对话框也可以创建一个 Access 数据库，只需单击 Create 按钮，它

将启动 Assess 数据库应用并允许新建数据库。单击 Repair 和 Compact 按钮可对数据库进行修复和压缩操作。

(3) System Database 一般设为 None, 如果特别为应用程序创建了一个系统数据库, 那么选择 Database, 并选择所需的 System Database。



图 1-6 “创建新数据源”对话框

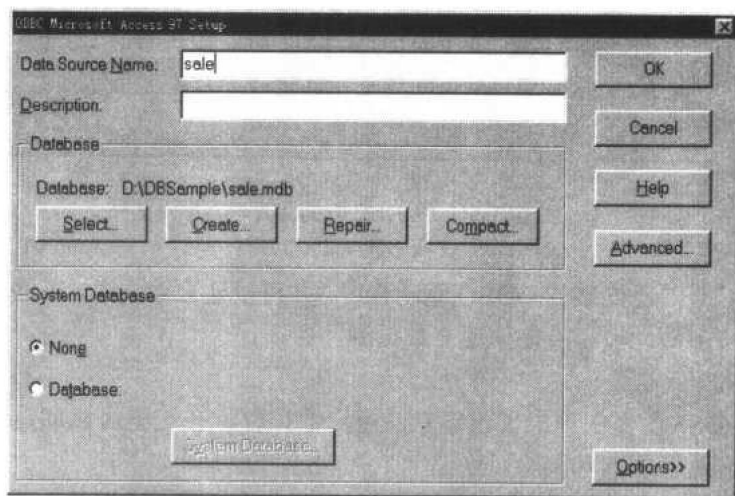


图 1-7 ODBC Microsoft Access 97 Setup 对话框

(4) 单击 Advanced 按钮, 会出现图 1-8 所示的 Set Advanced Options 对话框, 可以在 Login Name 和 Password 字段设置登录数据库的缺省用户名和口令。

(5) 在 ODBC Microsoft Access 97 Setup 对话框中单击 OK 按钮, 则在 ODBC 数据源管理器的用户 DSN 属性页上新增了一个名为 sale 的数据源, 如图 1-9 所示。