

● Practice...

Visual Basic 6.0

数据库编程思想与实践

Programming Series

谭小丹 刘国庆 李光明 编著

冶金工业出版社

Visual Basic 6.0 数据库编程思想与实践

谭小丹 刘国庆 李光明 编著

北 京

冶金工业出版社

2002

内容简介

本书以中文版 Visual Basic 6.0 为背景,从数据库编程基础知识、VisData、数据控件和绑定控件、DAO、ADO、数据环境设计器、报表制作、Client/Server 编程、RDO 编程等方面,详细介绍了 Visual Basic 6.0 数据库编程的基本操作和使用方法。本书最后以一个数据库运用进销存管理系统的开发为实例,阐述了该数据库的开发思路和步骤。本书实例和图表丰富、语言通俗易懂,在每章最后都附有一定数量的练习题以提高读者的实际操作能力。

本书适合于 Visual Basic 6.0 开发数据库应用程序的初学者,也可作为大专院校的计算机专业的教学用书,同时,对从事数据库开发的人员也是一本很好的参考书。

图书在版编目(CIP)数据

Visual Basic 6.0 数据库编程思想与实践 / 谭小丹等编

著. —北京:冶金工业出版社, 2002.8

ISBN 7-5024-3087-3

I. V... II. 谭... III. Basic 语言—程序设计

IV. TP312

中国版本图书馆 CIP 数据核字 (2002) 第 055774 号

出版人 曹胜利 (北京沙滩嵩祝院北巷 39 号, 邮编 100009)

责任编辑 程志宏

中山市新华印刷厂有限公司印刷; 冶金工业出版社发行; 各地新华书店经销

2002 年 9 月第 1 版, 2002 年 9 月第 1 次印刷

787mm × 1092mm 1/16; 25 印张; 579 千字; 391 页; 1-2500 册

40.00 元

冶金工业出版社发行部 电话: (010) 64044283 传真: (010) 64027893

冶金书店 地址: 北京东四西大街 46 号 (100711) 电话: (010) 65289081

(本社图书如有印装质量问题, 本社发行部负责退换)

前 言

1. 关于数据库和 Visual Basic

随着计算机在管理领域的广泛应用,计算机处理的数据量急剧增长。为了解决多用户、多应用共享数据的需求,出现了数据库技术。数据库无论对于公司还是个人而言,都是十分重要的。现代生活和数据库息息相关,没有现代的数据库,就没有现在我们所享有的生活方式,例如:银行依靠数据库来管理支票账目,从而提高了工作效率,而且在整个处理过程中很少出现错误。

随着数据库的应用开发,Visual Basic 成为很多程序员的首选编程语言,因为 Visual Basic 易学、易于调试,可以比其他编程语言更迅速地建立复杂的应用程序。此外,Visual Basic 中内置了许多用于开发数据库应用程序的专业化支持。程序员使用 Visual Basic 不仅可以开发传统的面向表格的数据库应用程序,还可以开发应用于 Web Server 的后台数据库,比使用传统的 Perl 脚本和 Active Server Pages 效率都要高。

2. 本书的特点

1) 组织合理,内容全面。

本书以中文版 Visual Basic 6.0 为背景,详细介绍了 Visual Basic 6.0 数据库编程的基本操作和使用方法,内容包括:VB 数据库编程基础知识、VisData、数据控件和数据绑定控件、DAO、ADO、数据环境设计器、报表制作、Client/Server 编程和使用 RDO。

2) 实例丰富,通俗易懂。

本书除介绍 Visual Basic 6.0 数据库编程基础理论外,还备有丰富的实例,这些实例涉及了程序设计的各个方面,既有简单的理论阐述,又有详细的制作过程,使读者能轻松而快速地使用 Visual Basic 6.0 编写数据库应用程序。为巩固对每章的学习,书中给出了一定数量的练习题,以便使读者在较短的时间内掌握 Visual Basic 6.0 数据库的编程。

3. 本书的内容结构

全书共分 10 章,各章内容如下:

第 1 章: Visual Basic 数据库编程基础。本章首先对数据库进行了简单的介绍,接着介绍了数据库的一些基本概念和术语、关系数据库设计、SQL 语言,以及 Visual Basic 和数据库。

第 2 章: Visual Basic 可视数据管理器。本章首先对 VisData 进行了介绍,然后介绍了打开 VisData 的方法、VisData 文件菜单、VisData 实用程序菜单、VisData 窗口和帮助菜单、使用 VisData 操作数据库。

第 3 章: Visual Basic 数据控件和数据绑定控件。本章主要介绍了应用数据控件访问数据库、Data 控件、ListBox 控件和 ComboBox 控件、DBList 控件和 DBCombo 控件、RemoteData 控件、ADO Data 控件,并分别给出了 RemoteData 控件和 ADO Data 控件实例。

第 4 章: 使用 DAO。本章主要从 DAO 基础知识、DAO 层次模型和编程中常用的几种对象三个方面,介绍了如何使用 DAO 3.51 版本,最后给出了一个使用 DAO 的编程实例。

第 5 章: 使用 ADO。本章首先对 ADO 进行了简单的介绍,然后介绍了 ADO 对象模型、ADOX 对象模型、ADO 的各个对象,最后通过一个实例介绍了如何通过 ADO 实现数

数据库观察器。

第 6 章：数据环境设计器。本章首先对数据环境设计器进行了简单的介绍，接着介绍了 Connection 对象、Command 对象，并通过实例介绍了数据环境设计器的使用。

第 7 章：报表制作。本章首先对报表设计器进行了概述，接着介绍了数据报表控件、报表设计器的使用、DataReport 对象、数据报表编程，并给出了数据报表编程实例。

第 8 章：Client/Server 编程。本章首先对 Client/Server 基础知识进行了介绍，接着介绍了 ODBC 的使用、使用 Visual Basic 访问数据、SQL Server 的配置与使用、使用 SQL Server 建立数据库对象，并给出了 Client/Server 编程实例。

第 9 章：使用 RDO。本章首先对 RDO 进行了概述，接着介绍了 RDO 对象模型、RDO 对象、RDO 编程，还介绍了 DAO、RDO、ADO 的区别，RDO 到 ADO 的转换。

第 10 章：综合实例——进销存管理系统的开发。本章通过一个进销存管理系统的具体开发过程，介绍了使用 Visual Basic 6.0 进行数据库开发的全过程。

本书每章后都有习题，书后还给出了参考答案，读者可以对照进行练习，以巩固所学的知识。

4. 本书的适合对象

本书适合于运用 Visual Basic 6.0 开发数据库应用程序的初学者，也可作为大专院校的计算机教材，同时，还可作为数据库开发人员的参考书。

本书主要由谭小丹、刘国庆、李光明编写，参加本书编写的还有卢智勇。编者在此表示深深地感谢。由于编者水平有限，书中不妥之处，恳请读者批评指正，编者将不胜感激。

编 者

2002 年 7 月

目 录

第 1 章 Visual Basic 数据库编程基础..... 1

1.1 数据库概述	1
1.1.1 数据库的发展	1
1.1.2 桌面数据库	2
1.1.3 分布式数据库	2
1.2 基本概念和术语	4
1.3 关系数据库设计	6
1.3.1 规划关系数据库	6
1.3.2 设计一个关系数据库	7
1.4 SQL 语言	9
1.4.1 SQL 的特点和基本功能	10
1.4.2 SQL 数据类型	11
1.4.3 SQL 语句	11
1.4.4 Select 语句	12
1.4.5 Insert 语句	12
1.4.6 Delete 语句	12
1.4.7 Update 语句	13
1.4.8 Create Table 语句	13
1.4.9 Create Index 语句	13
1.4.10 Create View 语句	13
1.5 Visual Basic 和数据库	13
1.5.1 Visual Basic 支持的数据库	14
1.5.2 Visual Basic 访问数据库的方案 ..	14
小结	18
综合练习一	18
一、选择题	18
二、填空题	18
三、思考题	19
四、上机实践	19

第 2 章 Visual Basic 可视数据管理器..... 20

2.1 VisData 概述	20
2.2 打开 VisData	20
2.2.1 创建一个示例数据库	21
2.2.2 在 VisData 中输入数据	24

2.2.3 VisData 工具栏按钮	26
2.3 VisData 文件菜单	28
2.3.1 打开数据库	28
2.3.2 新建	29
2.3.3 导入/导出	30
2.3.4 工作空间	31
2.3.5 错误	31
2.3.6 压缩 MDB	31
2.3.7 修复 MDB	32
2.4 VisData 实用程序菜单	32
2.4.1 查询生成器	32
2.4.2 数据窗体设计器	34
2.4.3 全局替换	35
2.4.4 附加	36
2.4.5 用户组/用户	37
2.4.6 SYSTEM.MD?	37
2.4.7 首选项	38
2.5 VisData 窗口和帮助菜单	38
2.6 使用 VisData 操作数据库	38
2.6.1 修改表结构	39
2.6.2 查询数据库	40
小结	40
综合练习二	41
一、选择题	41
二、填空题	41
三、思考题	42
四、上机实践	42

第 3 章 Visual Basic 数据控件

和数据绑定控件..... 43

3.1 应用数据控件访问数据库	43
3.2 Data 控件	45
3.2.1 Data 控件与数据相关的 主要属性	45
3.2.2 Data 控件与数据相关的方法	47

3.2.3 Data 控件与数据相关的主要事件	49	4.3.7 Field 对象	86
3.3 ListBox 控件和 ComboBox 控件	52	4.4 DAO 编程	88
3.4 DBList 控件和 DBCombo 控件	53	小结	93
3.5 RemoteData 控件	55	综合练习四	93
3.5.1 RemoteData 控件与数据相关的主要属性	55	一、选择题	93
3.5.2 RemoteData 控件与数据相关的方法	58	二、填空题	94
3.5.3 RemoteData 控件与数据相关的主要事件	59	三、思考题	94
3.5.4 RemoteData 控件举例	61	四、上机实践	94
3.6 ADO Data 控件	64	第 5 章 使用 ADO	95
3.6.1 ADO Data 控件与数据相关的属性	64	5.1 ADO 概述	95
3.6.2 ADO Data 控件与数据相关的方法	65	5.2 ADO 对象模型	97
3.6.3 ADO Data 控件与数据相关的事件	65	5.3 ADOX 对象模型	97
3.6.4 ADO Data 控件举例	68	5.4 ADO 对象	98
小结	72	5.4.1 Connection 对象	98
综合练习三	72	5.4.2 Recordset 对象	112
一、选择题	72	5.4.3 Field 对象	143
二、填空题	73	5.4.4 Command 对象	148
三、思考题	73	5.5 ADO 编程	152
四、上机实践	73	5.5.1 ADO 编程的步骤	152
第 4 章 使用 DAO	74	5.5.2 使用 ADO 实现一个数据库观察器	153
4.1 DAO 概述	74	小结	160
4.2 DAO 层次模型	74	综合练习五	161
4.2.1 DAO 层次模型概述	74	一、选择题	161
4.2.2 DAO 模型的层次结构	75	二、填空题	161
4.3 DAO 对象	76	三、思考题	161
4.3.1 DBEngine 对象	76	四、上机实践	161
4.3.2 Workspace 对象	78	第 6 章 数据环境设计器	162
4.3.3 Database 对象	79	6.1 数据环境设计器概述	162
4.3.4 TableDef 对象	81	6.2 Connection 对象	164
4.3.5 Recordset 对象	83	6.3 Command 对象	165
4.3.6 QueryDef 对象	85	6.3.1 设置 Command 对象的属性	166
		6.3.2 创建层次结构记录集	168
		6.3.3 创建合计数据	171
		6.4 使用数据环境设计器	173
		6.4.1 使用数据绑定控件创建窗体	176
		6.4.2 拖动对象创建数据窗体	176

6.5 数据环境设计器编程	177	四、上机实践	201
6.5.1 DataEnvironment 对象	177	第 8 章 Client/Server 编程	202
6.5.2 执行命令和操作记录集	179	8.1 Client/Server 基础知识	202
小结	180	8.1.1 基本概念	202
综合练习六	181	8.1.2 两层 Client/Server 模型	203
一、选择题	181	8.1.3 三层 Client/Server 模型	204
二、填空题	181	8.1.4 三层 Client/Server 的优点	205
三、思考题	181	8.2 ODBC 的使用	205
四、上机实践	181	8.2.1 ODBC 概述	205
第 7 章 报表制作	182	8.2.2 安装和配置 ODBC	206
7.1 报表设计器概述	182	8.3 使用 Visual Basic 访问数据	208
7.1.1 数据报表设计器的特性	182	8.4 SQL Server 的配置与使用	209
7.1.2 数据报表设计器的部件	182	8.4.1 SQL Server 概述	209
7.1.3 数据报表设计器的组成	183	8.4.2 SQL Server 的 Client/Server 特性	210
7.2 数据报表控件	184	8.4.3 SQL Server 数据库结构	212
7.2.1 RptTextBox 控件	184	8.4.4 SQL Server 数据类型	213
7.2.2 RptLabel 控件	185	8.5 使用 SQL Server 建立数据库对象	217
7.2.3 RptImage 控件	185	8.5.1 Enterprise Manager 简介	217
7.2.4 RptLine 控件	185	8.5.2 建立数据库	220
7.2.5 RptShape 控件	185	8.5.3 建立数据表	222
7.2.6 RptFunction 控件	185	8.5.4 管理安全性	224
7.3 报表设计器的使用	186	8.6 编程实例	225
7.3.1 使用数据环境设计器创建 数据源	186	小结	231
7.3.2 创建并显示数据报表	189	综合练习八	231
7.4 DataReport 对象	191	一、选择题	231
7.4.1 DataReport 对象的属性	191	二、填空题	231
7.4.2 DataReport 对象的方法	192	三、思考题	232
7.4.3 DataReport 对象的事件	195	四、上机实践	232
7.5 数据报表编程	196	第 9 章 使用 RDO	233
7.5.1 预览报表	196	9.1 RDO 概述	233
7.5.2 打印报表	198	9.2 RDO 对象模型	234
7.5.3 导出报表	198	9.3 RDO 对象	235
小结	200	9.3.1 rdoEngine 对象	235
综合练习七	200	9.3.2 rdoEnvironment 对象	240
一、选择题	200	9.3.3 rdoConnection 对象	244
二、填空题	200	9.3.4 rdoQuery 对象	251
三、思考题	201		

9.3.5 rdoResultset 对象	254	10.4.2 登录子菜单	306
9.3.6 rdoTable 对象	266	10.4.3 退出子菜单	308
9.3.7 rdoColumn 对象	268	10.5 数据库菜单	308
9.3.8 rdoParameter 对象	272	10.5.1 业务员数据表子菜单	308
9.4 RDO 编程	274	10.5.2 客户数据表子菜单	315
9.4.1 SQL Server 数据库的建立	274	10.5.3 厂商数据表子菜单	320
9.4.2 RDO 编程	275	10.5.4 商品数据表子菜单	324
9.5 RDO 与 ADO	284	10.5.5 进货数据表子菜单	327
9.5.1 DAO、RDO 和 ADO 的区别	284	10.5.6 销售数据表子菜单	331
9.5.2 RDO 到 ADO 的转换	285	10.5.7 退货数据表子菜单	335
小结	289	10.5.8 库存数据表子菜单	339
综合练习九	290	10.6 交易菜单	342
一、选择题	290	10.6.1 进货登记子菜单	342
二、填空题	290	10.6.2 销售登记子菜单	353
三、思考题	291	10.6.3 退货登记子菜单	362
四、上机实践	291	10.7 统计菜单	368
第 10 章 综合实例——进销存管理系统		10.7.1 统计结果显示窗体	368
的开发	292	10.7.2 各子菜单的编程实现	372
10.1 进销存管理系统功能简介	292	10.8 报表菜单	372
10.2 数据库设计	293	10.8.1 报表设计	373
10.2.1 建立实体模型	293	10.8.2 报表编程实现	375
10.2.2 创建数据库	295	10.9 窗口和帮助菜单	376
10.3 主界面和主菜单	296	10.9.1 窗口菜单	376
10.3.1 主界面设计	296	10.9.2 帮助菜单	376
10.3.2 模块编程	299	10.10 系统运行	377
10.3.3 frmMain 窗体代码	301	小结	383
10.4 用户菜单	306	参考答案	384
10.4.1 注销子菜单	306	参考文献	392

第 1 章 Visual Basic 数据库编程基础

随着数据库的应用开发, Visual Basic 成为很多程序员的首选编程语言, 因为 Visual Basic 易学、易于调试, 可以比其他编程语言更迅速的建立复杂的应用程序。此外, Visual Basic 中内置了许多用于开发数据库应用程序的专业化支持。程序员使用 Visual Basic 不仅可以开发传统的面向表格的数据库应用程序, 还可以开发应用于 Web Server 的后台数据库, 比使用传统的 Perl 脚本和 Active Server Pages 效率都要高。

在使用 Visual Basic 创建数据库应用程序之前, 必须对数据库有很好的了解, 并且知道 Visual Basic 如何访问数据库。在这一章中, 我们主要介绍数据库的发展、目前的几种数据库、数据库的一些基本概念和术语、设计数据库的规则、SQL 语言、Visual Basic 访问数据库的几种方案。

1.1 数据库概述

有多种多样的方法可以在计算机上存储信息, 但是最终都归结为把信息存储在文件中, 而文件放在磁盘驱动器上。根据存储方式的不同, 有两种类型的文件: 顺序存取文件和随机存取文件。顺序存取文件把文件当作一个单元存取, 随机存取文件允许用户访问文件的某一部分。

对于顺序文件中的数据进行操作, 必须读写整个文件, 传统的文件系统基本上属于这种方式。在大多数的应用程序中, 仅仅需要及时存取特定点的数据子集, 没有必要为了存取一小部分数据而对整个文件进行操作。为了实现对文件的随机存取, 随机存取文件把文件分成许多称为记录的块, 每个记录都有惟一的记录号, 记录号是根据记录在文件中的相对位置决定的, 可以根据记录号来读取或者更新一条记录, 于是有了数据库的雏形。

1.1.1 数据库的发展

从 20 世纪 60 年代后期以来, 随着计算机被广泛地应用于管理领域, 需要计算机处理的数据量急剧增长。为了解决多用户、多应用共享数据的需求, 出现了数据库技术。

数据库技术解决了在使用随机存取文件时所面临的以下几个问题:

- 1) 并发。指保证两个以上同时运行的程序在对数据进行操作时不会互相干扰。
- 2) 数据安全。指某一用户只能读写特定的文件, 而不能存取权限以外的其他文件。
- 3) 进行备份和使用事务日志。进行备份可保证在出现意外事故时数据的损失最小。事务日志是记录所有对数据的改变操作的文件, 仅包含对文件的修改内容。

数据库的出现使信息系统的研制从以加工数据的程序为中心转变到以共享的数据库为中心来进行。数据库开始是在大公司或机构中用来进行大规模事务处理的, 后来, 伴随着微机的普及, 数据库技术被应用到微机上, 出现了像 FoxBase、Access、FoxPro 这样的个人数据库应用。这类数据库系统也叫桌面数据库, 一般在单机环境下使用。

随着计算机数量的增加, 人们对软件和数据资源的共享越来越迫切, 传统的单机环境

下的数据库系统已不能满足用户的需求，用户希望能通过网络方便的跨部门、跨产业、甚至跨地区的访问数据资源。网络技术的迅猛发展，为网络数据库的产生奠定了基础。

网络数据库是以网络环境作为下层支撑的数据库系统的总称，一般有以下特点：

1) 数据库或数据库的处理可以被分布在网络中不同节点上完成。数据库的处理包括数据库的应用和数据库的服务两部分。

2) 数据资源能够在网络的各个节点中方便地传输和访问。

3) 能支持远程数据的同步查询和更新。

4) 具有必要的数据恢复能力。

5) 多用户系统。

网络数据库的发展离不开分布式数据库理论的提出，正是分布式数据库理论首次把计算机网络技术和数据库技术结合起来，才使得网络数据库的发展成为可能，并标志着网络数据库的产生。从 20 世纪 70 年代初分布式数据库技术的兴起，到如今基于客户/服务器体系结构的数据库、数据库的互联、互操作概念的相继提出，数据库一刻也离不开网络，并将一直和网络技术结合在一起。

1.1.2 桌面数据库

桌面数据库是一类数据库软件，有时也叫 ISAM (Indexed Sequential Access Method, 索引顺序存取方法) 数据库，因为这些数据库都使用 ISAM 文件。目前市场上有很多多种桌面数据库，例如，Microsoft Access、Microsoft FoxPro、Borland Paradox 等属于桌面数据库。这些数据库产品在很多方面都彼此不同，但作为桌面数据库，还是有以下一些共同点的：

1) 数据存放在 ISAM 文件中。

2) 数据文件是可自描述的。

3) 有自己的语言和数据类型。

4) 有自己的解释程序，可以运行自己语言编写的程序。

5) 提供标准的 DBMS (数据库管理系统) 功能。

6) 是专门为在个人计算机上运行而设计的。

7) 可以处理记录封锁，允许多个线程和应用程序同时访问数据。

8) 提供对记录中的数据类型检查功能，保证数据的完整性。

9) 提供基于集合的操作，一条命令可完成对成千上万条记录的处理。

桌面数据库的数据文件可以通过局域网供远程计算机访问，此时运行应用程序的计算机叫客户机，数据驻留的计算机叫服务器。当数据文件通过局域网访问时，数据要在客户机上进行处理，所有数据和索引文件都要从服务器一端通过网络传送到客户机进行处理，因此不适合建立高性能的客户/服务器应用系统。另外桌面数据库是为个人计算机设计的，其能力和数据吞吐量都是有限的，用户数量和数据文件的大小也有一定限制。

1.1.3 分布式数据库

分布式数据库是在集中式数据库的基础上，由计算机网络的引入而发展起来的新技术。可以简单的理解为：集中式的数据库是建立在一个单一计算机系统上的数据库，而分

布式数据库是建立在计算机网络上各个节点上的数据库的集成。分布式数据库一般指数据资源(物理库)在地理上存储在一个含有多个数据库管理系统的计算机网络的多个节点中,但是逻辑上把它们视为一个统一的逻辑数据库进行处理。即物理上的分布性和逻辑上的统一性。

与集中式数据库比较,分布式数据库有它本身的特点:

1) 节点自治。由于在分布式数据库中存在局部和全局两种应用,因此,对数据库的数据操作有两种控制,产生了全局和局部两种管理员。分布式数据库强调局部管理员的作用,即节点的数据由其局部管理员进行管理,节点间的协调工作也可由局部管理员来完成。

2) 数据独立性。分布式数据库使应用程序所面对的分布存放的数据库就像集中数据库一样,不必考虑它的分布性。

3) 数据冗余。由于节点自治概念的出现,分布式数据库的局部应用需要有一定的数据冗余度,用来提高系统可靠性,减少数据损失,以及增加数据复制,减少运行时间和通信费用。

4) 完整性、可恢复性、并发控制。分布式数据库从体系结构上讲,通常可分为综合型和联合型两种。综合型分布式数据库体系结构是指分布式数据库建立之前还没有建立独立的局部 DBMS,系统设计人员在综合权衡用户的需求之后,设计出一个完整的数据库管理系统,然后把系统的功能按照确定的方案选择,分散配置在这个分布式环境中。其中,每一个节点的数据库软件都谙熟整个网络的情况,也了解其他各节点的情况。从整体上看,它们组成了一个完整的系统,各个软件包之间的联系是靠网络存取进程间的通信服务来维持。这种体系结构如图 1-1 所示。

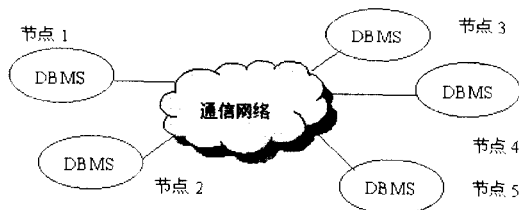


图 1-1 综合型体系结构

联合型的分布式数据库体系结构是以现存的集中式 DBMS 为基础,构成一个用分布式调制机制来完成分布处理功能的分布式数据库管理系统。联合型体系结构按各节点所保存的原数据库管理系统的异同,分为两种体系。如果每个节点的局部 DBMS 支持同一种数据模型、命令语言和查询语言,称为同构联合型体系结构,如图 1-2 所示。

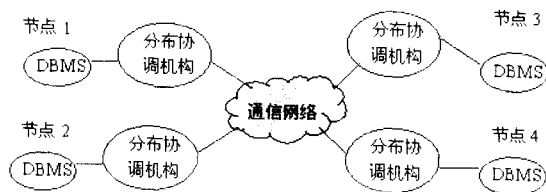


图 1-2 同构联合型体系结构

如果不同节点的局部 DBMS 支持的数据模型、命令语言和查询语言各不相同,称为异

构联合型体系结构,如图 1-3 所示。

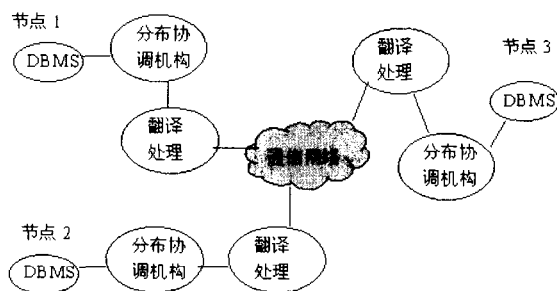


图 1-3 异构联合型体系结构

分布式数据库的查询处理和集中式数据库的查询处理相比较有两个特点,第一是由于查询所涉及的数据可能分布于网络不同的节点上,而节点间的数据要花费较多的时间,因此查询处理应考虑数据传送的开销。第二是由于各节点上的数据库可以独立的做一些查询处理,因此,查询处理应考虑如何让位于各节点的数据库并行的处理。

1.2 基本概念和术语

为了便于对数据库编程作全面的了解,在进入实际编程之前,我们先介绍一下在数据库编程中经常使用的一些概念和术语:

1. 数据库

对于早期的数据库系统,一个数据库文件就是一个二维数据表,随着数据库技术的发展,数据库所包含的内容不断得到扩展和延伸。对于现在我们所接触的大多数数据库系统来说,它已不仅仅是一张二维数据表的概念了,它存储的是若干二维数据表、数据表与数据表之间的关系和视图、触发器等众多内容。

2. 数据表

表用于存储数据,它以行/列方式组织,可以使用 SQL 从中获取、修改和删除数据。表是数据库的基本元素,数据库中一般有若干张表,而数据库本身则是管理这些表的。表有直观、方便和简单的特点,在我们生活中随处可见,如学籍表、统计表、月报表。数据表是以一组相关的数据行来排列,可以把它想象成行列整齐的表格,每一行称作一条记录,每一列称作一个字段。如表 1-1 所示为一个学生成绩表。

表 1-1 学生成绩表

学号	院系	班号	姓名	成绩
20000101	计算机系	计 2000	刘明	85
20000102	计算机系	计 2000	李敏	91
20000103	计算机系	计 2000	王文	88
20000104	计算机系	计 2000	赵辉	93

3. 临时表

临时表与数据表有着相似的性质,也可以看成是一张二维表格,对它的操作和数据表一样,不过它是只读的。临时表也是由一条条记录组成,而每条记录由若干个字段组成。

临时表是程序根据用户的需求查询相关数据表后生成的，往往在运行中存在于内存缓冲区中或以临时文件（.tmp 文件名）存在于磁盘上，运行完后自行释放。

4. 记录

记录指表中的一行，在一般情况下，记录和行的意思是相同的，它惟一的描述了一个有意义的事物。在表 1-1 中，每个学生所占据的一行是一条记录，描述了一个学生的情况。可以看出，在一个表中不允许有重复的记录，即每个记录都是惟一的。

5. 字段

字段是表中的一列，在一般情况下，字段和列所指的内容是相同的，它的每一个元素描述了该元素所在记录的某一方面性质。数据表是由其包含的各种字段定义的，每个字段描述了所包含的数据。在表 1-1 中，如“学号”一列即为一个字段。表中的每个字段都有一定的数据类型和变化范围，具体情况要视具体的数据库管理系统而定。每个表所能容纳的字段的数量，在不同的数据库管理系统中是不同的。字段可以包含各种字符、数字，甚至图形、声音、动画等多媒体信息。

6. 域

域是指字段的取值范围。如表 1-1 中对于“成绩”字段的取值范围为 0~100，任何一个学生记录的成绩信息都在这个范围之内。

7. 索引

索引是建立在表上的单独的物理数据库结构，基于索引的查询将使数据的获取更为快捷。索引可以是惟一的，也可以是不惟一的，主要看这些字段是否允许重复。主索引是表中的一列或多列的组合，作为表中记录的唯一标识。外部索引是相关联的表的一列或多列的组合，通过这种方式来建立多个表之间的联系。

8. 键

键就是数据表中为快速检索而被索引的一个字段或多个字段，键可以是惟一的，也可以是不惟一的，主要看这些字段是否允许重复。惟一键可以指定为主键，用来惟一标识数据表的每一行。如表 1-1 中的“学号”字段惟一的标识了一个学生，因此可以作为主键。

9. 关系

数据库可以由多个数据表组成，表与表之间可以用不同的方式相互关联。在数据库中，关系是建立在两个表之间的链接，以表的形式表现其间的链接，使数据的处理和表达有更大的灵活性。关系的概念虽然很简单但却很重要。在目前流行的数据库系统中，有三种关系：一对一关系、一对多关系和多对多关系。

10. 视图

视图是一个与真实表相同的虚拟表，用于限制用户可以看到和修改的数据量，从而简化数据的表达。是从一个或几个基本表导出的表，其本身不独立存储在数据库中。数据库中只存储视图的定义，不存储对应的数据，运行时系统自动将视图联接到相关的数据表。

11. 触发器

触发器是一组 SQL 命令的集合，当用户对表进行插入、删除或更新时触发自动执行。

12. 存储过程

存储过程是一个编译过的 SQL 程序，在该过程中，你可以嵌入条件逻辑、传递参数、

定义变量和执行其他编程任务。

13. 数据库引擎

一种管理数据怎样被存储和检索的软件系统。数据库的基本功能由数据库引擎提供。

1.3 关系数据库设计

DBMS 从产生发展到现在,出现了多种类型。按其数据模型来分,主要有层次数据库,网状数据库和关系数据库。前两种在 20 世纪 60~70 年代较为流行,然而它们表示数据之间的联系太过于复杂,现在已经很少使用。只是在讲解数据库的时候才有所提及。

关系数据库模型是由 Ted Codd 于 1970 年提出的,他认为数据库应当以表的形式(即关系)组织数据。关系数据库以关系的数学理论为基础,是数据库技术的一项重大突破。

在关系数据库中,用户可以不关心数据的存储结构,与此同时,关系数据库的查询可以用高级语言来描述,这大大提高了查询效率。正是由于关系数据库的简单严谨的特点,它在过去几十年来有了长足的发展,现在已经成为事实上的标准,当今,几乎所有的数据库产品都是基于关系数据库的。无论是 Access, FoxPro, dBase, 还是开放数据库互连(ODBC)产品如 SQL Server、Oracle 等数据库管理系统都是关系数据库系统。其中的开放数据库互连产品如 SQL Server、Oracle 还可以和计算机网络一起组成分布式数据库系统。

数据库设计是数据库应用的一个关键因素,设计结构合理、功能齐全的数据库对于提高数据库应用程序的开发效率和程序的性能都是非常重要的。数据库设计是指对于一个给定的应用环境,构造最优的数据模式,建立数据库,使其能够有效地存储数据记录,并能满足各种应用需求。数据库的设计通常是以一个已经存在的数据库管理系统为基础的,在设计一个数据库时,应该注意将数据库的设计与应用系统的设计结合起来,也就是说,要注意把结构(数据)设计和行为(处理)设计结合起来。数据库设计质量的好坏将直接影响到系统中各个处理过程的质量和运行性能。一个设计失败的数据库往往到了应用程序的开发阶段还要不断地修改。

1.3.1 规划关系数据库

进行数据库的规划是所有数据库编程的第一步,也是最重要的一步。一个经过精心设计的数据库可使你非常方便地访问它,可保证你在后期的程序设计中能够随心所欲的存取任何你所关心的数据。因此,在进行正式编程之前,一定要花足够的精力来进行数据库的规划。这里所说的规划主要指以下几件事情:

首先,列出客户所关心的所有数据和建立这个数据库的主要目的。这可以通过收集各种相关的数据报表和表格来完成。在此期间,一定要经常和客户保持沟通,收集的数据一定要完整,每个数据的含义要尽可能的理解。需求分析是数据库设计的第一阶段,不断的调查与研究,了解组织机构的情况。了解部门的业务流程等系统需求,对于设计好数据库是非常重要的。在需求分析中,应该对现实世界要处理的对象进行详细地调查,在确定系统功能的过程中,收集支持系统功能目标的数据以及分析数据之间的关系和处理流程。

然后,就可以进行实质性的数据表规划工作了。一般来说,进行数据库的数据表规划是整个数据库设计中技巧性最强的一个步骤。因为此时虽然已经有了一些数据库的轮廓感

觉,但所搜集的数据只是反应了客户自己所关心的数据格式,而在数据库中只能存放数据库特有的数据格式。因此,要把收集到的数据信息转化为与所采用的数据库管理系统所支持的数据模型相符合的数据形式。对于数据表的规划,一种行之有效的方法是对各个数据信息进行仔细研究,确定每种数据信息的实际内容,然后把具有同一个主题的数据规划到一个表中。不相干的信息不要放到同一个表中,以免造成数据库存储信息的冗余。

在确定了数据表后就可以对数据表中的字段进行规划,确定每个数据表中包含哪些字段。对于一个数据表来说,一般总有一个主题,那么该数据表的字段就围绕这个主题来确定。

规划数据库要完成以下三个目标:

- 1) 消除包含在表中的冗余信息。
- 2) 适应表结构在未来发生的变化。
- 3) 在处理数据的前端应用程序的数据库结构发生变化时,使可能发生的冲突最小。

规划数据库有以下三条基本原则:

1) 数据库中的记录个数应该反映现实世界对象的个数。如果一个对象的实例在现实世界中存在,则在数据库中有且只有一条记录存在。例如,要储存客户的信息,则对应每一个客户,在数据库中都是一条记录,且仅有一条。一条记录不能包括多个客户的信息,也不能将一个客户的信息分解在多条记录中存放。该原则是设计关系数据库的基础,可以使应用程序更有价值,因为它可以保证数据库的开放性和可访问性。

2) 每行记录的字段应该表达现实世界对象的属性。

3) 现实世界对象之间的关系应该反映在数据库记录之间的关系上。

规划数据库应该应用以下三个范式:

- 1) 在表中没有重复的组。
- 2) 表中非键字段都必须完全依赖于主键,如果主键是一个符号键,就必须依赖于主键中的每一个字段。
- 3) 索引非键字段都必须依赖主键,且非键字段间彼此要独立。

1.3.2 设计一个关系数据库

前面学习了一些在设计数据库时用到的一些原则和方法,现在我们将结合一个实例来介绍一下如何设计一个数据库。一般设计一个数据库可分以下几个步骤进行:

1. 陈述问题

陈述问题看起来简单,实际做起来却很难。陈述问题要表现出对体系结构的理解,同时着重于商业需要。如果是替换现有的数据库,就要把现有的数据库作为解决问题的基础,如果是第一次建立数据库,就要理解所要解决的问题。

如果连问题都提不出来,就谈不上解决问题。陈述问题应当尽可能简单。对于我们要讨论的实例,我们可以这样描述问题:对某公司的销售员和客户的基本情况和订单情况进行管理。

2. 分析问题

一旦对要解决的问题有了基本的了解,就可以建立数据来存储相关的信息。我们对提

出的问题进行分析，可以得到具体的信息。

首先要管理客户的基本情况，就必须建立一个客户信息表，在这个数据表中我们存储客户的基本信息，如客户姓名、性别、客户编号、年龄、电话、地址等信息。

其次要管理销售人员的基本资料，就要建立一个销售人员的信息表，存储销售人员的基本信息，如销售员姓名、性别、编号、年龄、电话和地址等。

最后，要对订单情况进行管理，还要建立一个订单信息表，记录公司的每一笔订单信息，如订单编号、订单金额、客户编号、销售员编号、订单日期等。

3. 建立实体和关系模型

实体是可以被惟一确定的，例如，一个客户、一个销售员或者一份订单就是一个实体。每个实体都有一系列属性，用来描述实体。例如，每个客户都有名字、住址等属性。关系产生在两个实体之间，例如，客户和订单的关系。

通过前面的分析，可以建立客户、销售员和订单的实体模型如表 1-2、表 1-3 和表 1-4 所示。

表 1-2 客户信息实体模型

字段名	数据类型	长度	描述
Customer_id	Text	12	客户编号
Customer_name	Text	10	客户姓名
Customer_sex	Text	2	客户性别
Customer_age	Integer	2	客户年龄
Customer_phone	Text	12	客户联系电话
Customer_addr	Text	40	客户联系地址

表 1-3 销售员信息实体模型

字段名	数据类型	长度	描述
Saler_id	Text	12	销售员编号
Saler_name	Text	10	销售员姓名
Saler_sex	Text	2	销售员性别
Saler_age	Integer	2	销售员年龄
Saler_phone	Text	12	销售员联系电话
Saler_addr	Text	40	销售员联系地址

表 1-4 订单信息实体模型

字段名	数据类型	长度	描述
Order_id	Text	12	订单编号
Customer_id	Text	12	客户编号
Saler_id	Text	12	销售员编号
Order_amount	Currency	-	订单金额
Order_date	Date	8	订单日期

创建了实体模型之后，现在来确定实体的关系。前面说过有三种基本的关系：一对一、一对多和多对多。这些关系指一个实体中与另外实体中的实例相关的实例数目。在一对一关系中，一个实体中只有一个实例与另一个实体中的一个实例相关。在一对多关系中，一