北京科海培训中心

Visual FoxPro
管理信息系统设计与实现

姬景刚　黄青云　张洪波　编著

科学出版社
2000

内 容 简 介

本书作者根据长期使用Visual FoxPro进行数据库开发和管理的实践经验，通过一个管理信息系统实例——教务管理信息系统，详细介绍了使用Visual FoxPro开发管理信息系统的全过程。书中穿插讲述了Visual FoxPro程序编写的基本方法和相关知识，并且讲解了数据库和信息管理的基本原理和知识。

通过本书的学习，读者可以掌握使用Visual FoxPro编写应用程序的一般方法，并且可以掌握如何使用Visual FoxPro开发管理信息系统。

本书内容详尽、图文并茂，并附以大量示例，可作为广大数据库管理人员和自动化办公人员的培训教程，也可供各类管理信息系统的开发和维护人员参考。

图书在版编目(CIP)数据

Visual FoxPro管理信息系统设计与实现/姬景刚等编著
—北京 科学出版社，2000 4

ISBN 7-03-008446-2

Ⅰ V… Ⅱ 姬 Ⅲ 关系数据库-数据库管理系统，
Visual FoxPro 6 0 Ⅳ TP311 138

中国版本图书馆CIP数据核字(2000)第06470号

科学出版社 出版
北京东黄城根北街16号
邮政编码 100717

北京门头沟胶印厂印刷

新华书店北京发行所发行 各地新华书店经售

*

2000年4月第 一 版 开本 787×1092 1/16
2000年5月第二次印刷 印张 21⅜
印数 5001 - 10000 字数 522 000

定价：30 00元

前　言

Visual FoxPro 是微软公司开发的数据库管理系统，它操作简单、功能强大，深受我国广大数据库开发人员的喜爱。Visual FoxPro 是 Windows 下最流行的桌面数据库管理系统之一，也是目前最简单实用的数据库管理系统。Visual FoxPro 提供了数据库设计器、表单设计器、查询设计器、菜单设计器和报表设计器等许多方便快捷的可视化操作工具，还提供了数据库向导、表向导、报表向导、查询向导以及表单向导等，便于用户迅速创建出所需的管理信息系统，并利用 Visual FoxPro 管理数据库系统。

本书从基本操作入手，结合具体的管理信息系统示例，全面地介绍了使用 Visual FoxPro 开发管理信息系统的过程和方法。

在本书的示例和附图中，使用的都是 Visual FoxPro 6 0 版本的界面，如果读者使用 Visual FoxPro 3 0 或 5 0，界面可能会有所不同。但本书的主要目的并不在于介绍 Visual FoxPro 6 0，而是以该版本为例介绍使用 Visual FoxPro 开发管理信息系统应掌握的知识，因此，不会给使用其他版本的读者带来困难。由于 Visual FoxPro 6 0 是当前该类型数据库产品的最新版本，其功能比以往的任何版本都更强大，因此本书推荐读者使用 Visual FoxPro 6 0。

本书作者长期使用 Visual FoxPro 进行数据库开发与管理，在实践中积累了丰富的经验和心得，能深刻体会读者使用 Visual FoxPro 进行管理信息系统开发时常遇到的疑难问题，因此在讲解基本操作和开发知识的同时，也穿插讲述一些常见问题的解决方案。

本书由潇湘工作室总策划，姬景刚主编。参加本书编写的人员还有黄青云、张洪波、晋成周、江新会、董立伟、韩金民、刘军等，数据库高级开发人员贺民对本书进行了检查和实例代码的测试，也有许多人在本书的校对、排版过程中付出了辛勤的汗水，在此一并表示感谢。正是这种群策群力的合作，这本书才能得以很快出版，由于时间仓促，加之水平有限，书中的缺点错误在所难免，恳请广大读者批评指正。

目 录

第1章 管理信息系统简介 1

1.1 信息与数据 1
1.2 管理信息系统 2
1.3 关系数据库概述 2
1.3.1 一对一关系 3
1.3.2 一对多关系 3
1.3.3 多对多关系 4
1.4 管理信息系统的基础技术 6
1.4.1 数据库技术 6
1.4.2 计算机网络技术 6
1.4.3 安全保密技术 7
1.5 管理信息系统开发方法 8
1.5.1 开发策略 8
1.5.2 开发方法 9

第2章 Visual FoxPro数据库开发环境 11

2.1 Visual FoxPro的发展历程 11
2.1.1 FoxBase 11
2.1.2 FoxPro 1.0 11
2.1.3 FoxPro 2.0 12
2.1.4 FoxPro 2.5 13
2.1.5 Visual FoxPro 3.0 13
2.1.6 Visual FoxPro 5.0 13
2.1.7 Visual FoxPro 6.0 14
2.2 Visual FoxPro中数据库的概念 14
2.2.1 表 14
2.2.2 联接 15
2.2.3 视图 15
2.2.4 存储过程 15
2.3 Visual FoxPro 6.0的新特点和新功能 15
2.4 Visual FoxPro用户界面 17
2.4.1 系统菜单 18
2.4.2 工具栏 30

2.4.3 命令窗口.....31
2.5 Visual FoxPro的快速设计工具——向导.....32
2.6 Visual FoxPro的主要设计工具——设计器.....34
2.7 Visual FoxPro的辅助设计工具——生成器.....34
第3章 建立管理信息系统项目.....36
3.1 “教务管理信息系统”概况和开发步骤.....36
3.2 使用项目管理器.....36
3.2.1 项目管理器窗口.....37
3.2.2 设置项目中的文件.....38
3.3 添加数据库.....40
3.4 创建用户界面.....40
3.4.1 系统菜单.....40
3.4.2 输入输出表单.....41
第4章 建立教务管理信息数据库.....42
4.1 数据库设计步骤.....42
4.2 创建数据库.....42
4.3 添加表.....44
4.3.1 表的基本知识.....44
4.3.2 添加表.....49
4.4 使用索引.....55
4.4.1 索引的概念.....55
4.4.2 索引类型.....56
4.4.3 建立索引.....57
4.5 建立表之间的关系.....60
4.6 使用数据字典.....63
4.6.1 长表名.....64
4.6.2 表的注释.....64
4.6.3 长字段名.....65
4.6.4 字段的注释.....65
4.6.5 字段的标题.....65
4.6.6 指定输入掩码.....65
4.6.7 定义格式.....66
4.6.8 设置默认类.....67
4.6.9 字段默认值.....67
4.6.10 字段级规则.....69
4.6.11 使用触发器.....71
4.7 修改记录.....73

4.7.1 追加记录.....73
4.7.2 引入记录.....74
4.7.3 删除记录.....75
4.8 筛选表.....76
4.9 数据库的备份与恢复.....77
4.9.1 普通数据库的备份与恢复.....78
4.9.2 大型数据库的备份与恢复.....78
第5章 创建数据查询和视图 79
5.1 SQL标准查询语句.....79
5.2 使用查询向导.....82
5.3 使用查询设计器.....86
5.3.1 查询设计器界面.....86
5.3.2 选择查询字段.....88
5.3.3 设定查询条件.....89
5.3.4 设置排序依据.....91
5.3.5 对查询结果分组.....92
5.3.6 控制记录输出.....95
5.4 输出查询结果.....96
5.5 设计视图.....97
5.5.1 创建本地视图.....97
5.5.2 创建远程视图.....101
5.6 利用视图更新数据.....102
5.6.1 更新表.....102
5.6.2 设置关键字段.....103
5.6.3 创建远程数据源规则.....103
5.6.4 更新指定字段.....104
5.6.5 控制如何检查更新冲突.....104
5.6.6 更新数据源.....105
第6章 Visual FoxPro程序设计基础..... 107
6.1 Visual FoxPro的编程机制.....107
6.1.1 使用命令窗口.....107
6.1.2 编写程序文件.....108
6.1.3 使用代码窗口.....109
6.2 Visual FoxPro 6.0的数据类型.....109
6.3 存储数据.....111
6.3.1 常量.....111
6.3.2 变量.....111

6.3.3 数组 112
6.3.4 对象的属性 113
6.4 处理数据 113
6.4.1 操作符 113
6.4.2 命令 115
6.4.3 函数 115
6.5 流程控制语句 116
6.5.1 顺序语句 116
6.5.2 条件语句 116
6.5.3 循环语句 117
第7章 面向对象的事件驱动程序设计 119
7.1 对象和类 119
7.2 类的特性 120
7.2.1 封装性 120
7.2.2 继承性 120
7.2.3 多态性 121
7.3 Visual FoxPro中的基类 121
7.3.1 容器类 121
7.3.2 控件类 122
7.4 对象操作 123
7.4.1 创建对象 123
7.4.2 引用对象 124
7.4.3 设置属性值 125
7.5 创建自定义类 125
7.5.1 直接编码创建对象类 125
7.5.2 用类设计器创建对象类 129
7.5.3 把表单存储为类 131
7.6 创建教务查询类 132
7.7 注册类库 134
7.7.1 在“选项”对话框中注册 134
7.7.2 在“表单设计器”中直接注册类库 136
7.8 常用属性、事件和方法 136
7.8.1 常用属性 136
7.8.2 常用事件 137
7.8.3 常用方法 138
7.9 设计面向对象的程序 138
第8章 设计教务管理系统菜单 148

8.1 使用菜单设计器.....148
8.1.1 菜单名称.....149
8.1.2 结果.....149
8.1.3 选项.....151
8.1.4 菜单级.....152
8.1.5 菜单项.....152
8.1.6 预览.....152
8.2 规划菜单系统.....152
8.2.1 规划菜单栏.....153
8.2.2 设计子菜单.....153
8.2.3 给菜单项排序、分组.....155
8.2.4 指定菜单项名称和提示信息.....156
8.2.5 指定快捷键与键盘访问键.....157
8.2.6 生成菜单程序.....158
8.3 添加菜单所需的代码.....159
8.3.1 添加初始化代码.....159
8.3.2 添加共用过程.....161
8.3.3 给菜单项指定过程.....161
8.3.4 建立默认过程.....163
8.3.5 添加清理代码.....164
8.4 动态控制菜单.....164
8.4.1 控制菜单的启止状态.....164
8.4.2 控制菜单是否复选.....165
8.5 创建快捷菜单.....166
8.5.1 设计快捷菜单.....166
8.5.2 在表单中使用快捷菜单.....168
8.6 设计自定义工具栏.....168
8.6.1 设计自定义工具栏类.....169
8.6.2 向表单集中添加自定义工具栏.....171
第9章 设计教务管理系统界面.....172
9.1 使用表单向导.....172
9.2 使用表单设计器.....175
9.2.1 设置数据环境.....176
9.2.2 添加控件.....177
9.2.3 设置对象属性.....189
9.2.4 添加新属性.....191
9.2.5 编写响应事件的代码.....193
9.3 创建学生成绩表单.....195

9.4 创建数据维护表单.....199
9.5 创建数据统计表单.....204
9.5.1 个人成绩统计.....204
9.5.2 课程统计.....209
9.5.3 班级统计.....213
9.6 创建学生选课表单.....218
9.7 创建报表.....226
9.7.1 报表向导.....226
9.7.2 报表设计器.....229
9.7.3 输出报表.....235
9.7.4 其他报表.....238
9.8 创建打印报表的表单.....239
9.8.1 打印学生课表.....239
9.8.2 打印课程成绩.....242
9.8.3 打印班级信息.....244
第10章 调试应用程序.....247
10.1 错误类型.....247
10.1.1 语法错误.....247
10.1.2 逻辑错误.....249
10.1.3 表达式错误.....249
10.2 使用调试器.....250
10.2.1 调试器工具栏.....251
10.2.2 跟踪窗口.....251
10.2.3 设置断点.....253
10.2.4 局部窗口.....255
10.2.5 监视窗口.....256
10.2.6 调用堆栈窗口.....256
10.2.7 调试输出窗口.....257
10.2.8 事件跟踪.....257
10.3 运行时刻错误的处理.....258
10.3.1 处理错误的函数与命令.....258
10.3.2 避免错误.....259
10.3.3 处理过程错误.....259
10.3.4 处理类和对象中的错误.....260
10.3.5 从错误处理代码中返回.....262
第11章 集成应用程序项目.....263
11.1 构造应用程序框架.....263

11.2 建立主控程序 265
11.3 设置项目主文件 266
11.4 将文件加入到项目 266
11.5 引用可修改的文件 267
11.6 编译应用程序 268
11.6.1 测试项目 268
11.6.2 连编应用程序 269
第12章 发布应用程序 271
12.1 考虑使用者的软硬件环境 271
12.2 处理运行时的错误 271
12.3 保护源代码 272
12.4 包含资源文件 272
12.5 重新连编项目 273
12.5.1 设置项目信息 273
12.5.2 设置版本信息 275
12.5.3 连编为独立运行程序 275
12.6 发布准备 276
12.6.1 建立发布目录 276
12.6.2 添加支持文件 276
12.7 创建发布磁盘 277
第13章 多媒体 282
13.1 使用ActiveX绑定型控件 282
13.2 调用Windows的动态链接库文件 288
13.2.1 声明库函数（DECLARE-DLL命令） 288
13.2.2 调用API函数 289
13.2.3 调用Windows API控制多媒体 289
第14章 常用命令与函数 294
14.1 命令 294
14.1.1 面向对象的编程命令 294
14.1.2 环境设置命令 296
14.1.3 数据库操作命令 300
14.1.4 数据共享命令 309
14.1.5 键盘和鼠标命令 311
14.2 函数 313
14.2.1 数据库操作函数 313
14.2.2 字符函数 318

14.2.3 数据转换函数.....322
14.2.4 SYS()函数.....324
14.2.5 文件管理函数.....326

第 1 章　管理信息系统简介

信息、材料和能源并称为当代社会发展的三大支柱，人们对信息的利用程度已经成为衡量社会文明程度和科技发展水平的重要指标。信息管理系统正是为了适应信息技术的需要而发展起来的，它是信息科学、管理科学和计算机科学等相结合而形成的一门新兴综合学科，已发展成为计算机应用中的重要领域。根据现代经营管理模型建立起来的管理信息系统，利用计算机对管理领域采集的各种信息进行加工，转化为数字信息并储存起来，然后根据需要进行查询、分析、汇总和更新。

管理信息系统最早出现于20世纪60年代，经过几十年的发展，目前已经拥有比较成熟的技术，本章中将介绍这方面的一些基础知识。

1.1　信息与数据

信息管理系统的处理对象是信息，而现实生活中的信息与计算机所能处理的数据是不同的，计算机处理数据后，还要以信息的形式把数据处理的结果反馈出来。

信息与数据是两个密不可分的概念：数据一般指用以记录信息的物理符号，如数字、字符串等，它可以直接储存在计算机外存中；信息是现实生活中人们直接使用的信号，它直接用来反映客观事物规律和特点，通常是直接从社会生活中采集数据然后加工得来的。

实质上，信息管理系统就是实现信息与数据直接的转化和处理，信息管理系统、数据和信息这三者的关系很像调制解调器、数字信息和模拟信息之间的关系。计算机处理的是数字信息，电话线路能够传输模拟信息，调制解调器实现两者之间的转化。在上载时，调制解调器把数字信息调制为模拟信息，然后通过电话线路传送出去；在下载时，调制解调器把从电话线路中传送来的模拟信息解调为数字信息，然后由计算机阅读和处理。信息管理系统的功能与调制解调器的作用很相似，它通过计算机把客观世界的信息处理为能被计算机处理的数据，在计算机中存储起来，而当用户需要查询这些信息时，信息管理系统便把存储的数据进行反向运算，转换为人们熟知的信息，并显示出来。这三者的关系可以用图1-1来表示。

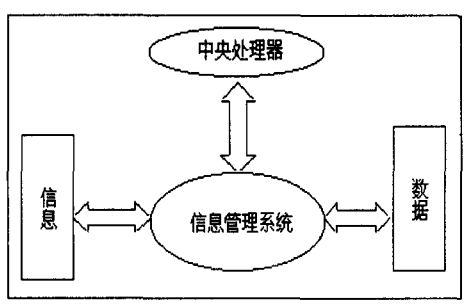


图 1-1　数据、信息和管理信息系统的关系

1.2 管理信息系统

管理信息是指能对生产经营活动产生影响的数据，这些数据是对各种日常生产经营活动过程中收集的数据进行加工处理、分析解释后形成的。

管理信息系统又称MIS，是英文Management Information System的简写。管理信息系统是一个由用户和计算机（包含硬件和相应软件）组成的进行管理信息收集、存储、处理、传递、使用和维护的软件系统。它建立在现代管理科学、系统工程、数值方法和计算机科学等多种前沿科学基础之上，能够代替管理人员进行复杂的数据统计、数据分析、事务处理，并提供决策方案，从而帮助企事业单位提高工作效率，并节约开支。

常用的管理信息系统有人事管理信息系统、仓储管理信息系统、财务管理信息系统和设备管理信息系统等。本书以一个完整的教务管理系统为例，结合Visual FoxPro数据库开发系统，来讲述创建各种数据文件和用户界面的方法，最终建立一个完整的管理信息系统。

1.3 关系数据库概述

长期以来，管理信息系统一直用于存储、组织、处理和维护大量的数据记录，其主要对象就是数据库。用户可以用多种方法组织和查询数据库中存储的数据。当前流行的数据库模型有层次型、关系型和网络型，而关系型数据库由于其使用简单，而且技术成熟，是当前数据库模型的主流，许多著名的数据库开发系统都使用这种模型，如FoxBase、FoxPro、Oracle和Access等。

虽然使用计算机能够快速处理数据，但大量繁杂无序的数据记录堆积在一起，不仅会造成数据资源的浪费，而且还影响用户查询和处理信息的效率。为此，在20世纪70年代，E.F.Codd发明了一种新的方法，把各种数据结合在一起，这种方法通过分析各种数据之间的关系，把结构基本相同的记录组合在一起，不同的结构则分别存放起来，然后，用公共关键字段把不同结构的记录联系起来，这样就能尽可能减少数据重复，这就是著名的关系数据库模型。例如，在教务管理中，学校要掌握每个学生的基本信息，如学号、姓名、性别、家庭住址、电话号码等，还要掌握各门课程的课程编号、课程名称、上课地点、上课时间、学分数、课程类型和主讲教师等信息，如果把这两种结构不同的信息存储在同一个表中，就会造成数据的大量重复，而使用关系型数据库模型，就可以把这些信息结合在两个不同的表中，通过学号和课程编号两个主关键字段把这两个表联系起来。

在关系型数据库中，一个数据库是由许多表组成的，这些表结构不同但相互联系。然而，不同的表之间的关系是不同的，也就是说，它们之间的连接方式是不同的，可以分为下面的三种：

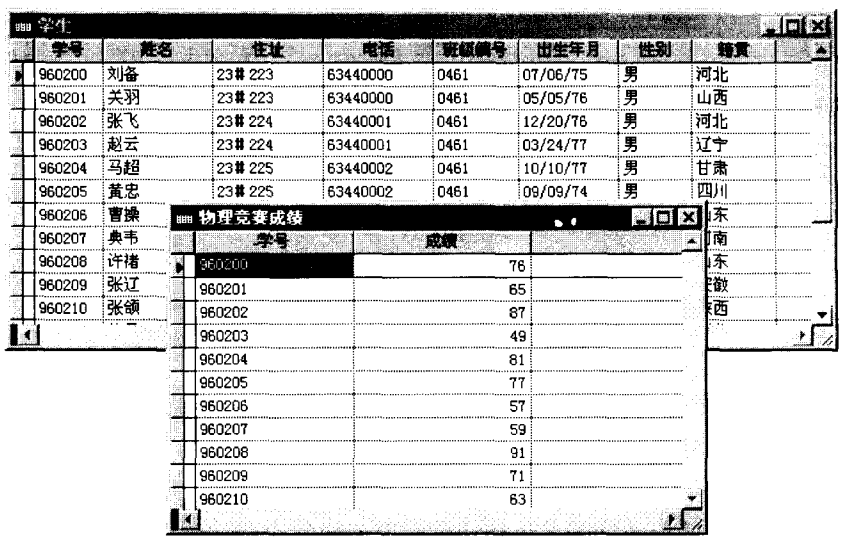
- 一对一关系
- 一对多（多对一）关系
- 多对多关系

关系数据库用主关键字段按照这些关系将各种数据联系起来，下面分别介绍这三种关系。

1.3.1　一对一关系

在一对一关系中，表A的一个记录在表B中只能对应一个记录，而表B中的一个记录在表A中也只能有一个记录与之对应。

两表间的一对一关系不经常使用。这是因为，在许多情况下，两个表的信息可以简单地合并成一个表，通常，在创建临时表的情况下，才使用这种关系。例如，在教务管理信息系统中，已经创建了一个“学生”表，现在需要建立一个临时表“物理竞赛成绩”，如图1-2所示。“学生”表中的一个学生记录对应于“物理竞赛成绩”表中一个成绩的记录，不同的学号对应于不同的成绩记录，这两个表的关系就是一对一关系。



学生

学号	姓名	住址	电话	班级编号	出生年月	性别	籍贯
960200	刘备	23# 223	63440000	0461	07/06/75	男	河北
960201	关羽	23# 223	63440000	0461	05/05/76	男	山西
960202	张飞	23# 224	63440001	0461	12/20/76	男	河北
960203	赵云	23# 224	63440001	0461	03/24/77	男	辽宁
960204	马超	23# 225	63440002	0461	10/10/77	男	甘肃
960205	黄忠	23# 225	63440002	0461	09/09/74	男	四川
960206	曹操						东
960207	典韦						南
960208	许褚						东
960209	张辽						徽
960210	张颌						西

物理竞赛成绩

学号	成绩
960200	76
960201	65
960202	87
960203	49
960204	81
960205	77
960206	57
960207	59
960208	91
960209	71
960210	63

图 1-2　两个一对一关系的表

如果数据库中一对一关系的两个表都不是临时表，就要考虑是否能把这些信息合并到一个表中。例如，如果要把物理竞赛成绩永久保留，可以在“学生”表中添加一个“物理竞赛成绩”字段，把学生的物理竞赛成绩作为记录添加到这个字段中，即把这两个表合并为一个表。如果出于某种原因不想这样做，还可按下面的方法建立一对一关系：

如果两个表有同样的主题，可在两个表中使用同样的主关键字字段，并以此建立一对一关系。

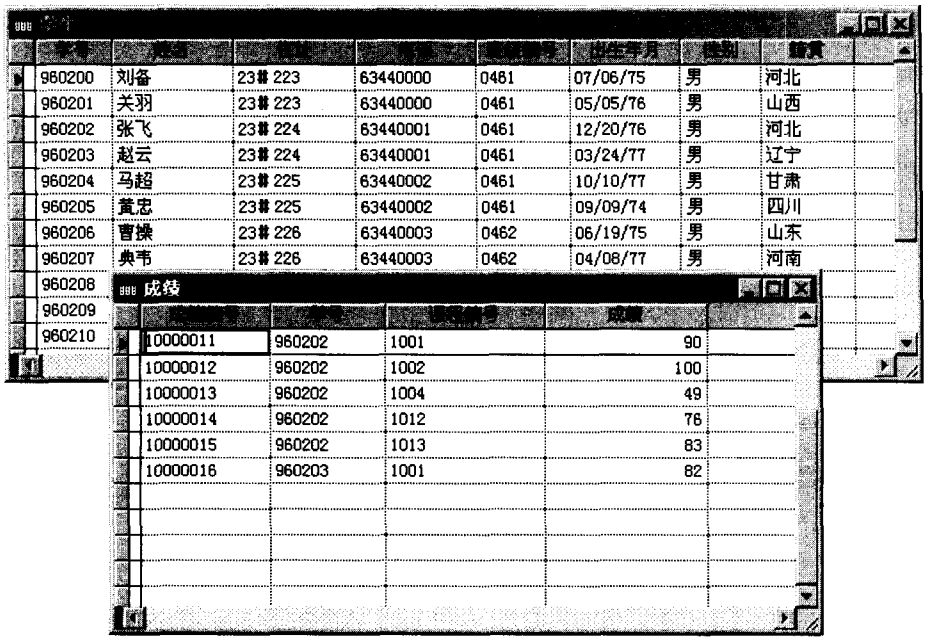
如果两个表有不同的主题及不同的主关键字，可选择一个表（随便哪个），把它的主关键字段放到另一个表中作为外部关键字。

1.3.2　一对多关系

一对多关系是关系型数据库中最普通的关系。在一对多关系中，表A的一个记录在表B中可以有多个记录与之对应，但表B中的一个记录最多只能有一个表 A 的记录与之对应。

例如，教务信息管理系统数据库中的“学生”表和“成绩表”就是一对多的关系。如图1-3所示，在“学生”表中一个学生有多门课的成绩，但是一个成绩记录只属于一个学

生，所以它们是一对多的关系。



学号	姓名	班级	电话	家庭住址	出生年月	性别	籍贯
960200	刘备	23班 223	63440000	0461	07/06/75	男	河北
960201	关羽	23班 223	63440000	0461	05/05/76	男	山西
960202	张飞	23班 224	63440001	0461	12/20/76	男	河北
960203	赵云	23班 224	63440001	0461	03/24/77	男	辽宁
960204	马超	23班 225	63440002	0461	10/10/77	男	甘肃
960205	黄忠	23班 225	63440002	0461	09/09/74	男	四川
960206	曹操	23班 226	63440003	0462	06/19/75	男	山东
960207	典韦	23班 226	63440003	0462	04/08/77	男	河南
960208							
960209							
960210							

成绩

成绩号	学号	课程号	成绩
10000011	960202	1001	90
10000012	960202	1002	100
10000013	960202	1004	49
10000014	960202	1012	76
10000015	960202	1013	83
10000016	960203	1001	82

图 1-3 具有一对多关系的两个表

要建立这样的关系，就要把关系中“一方”的主关键字字段添加到“多方”的表中。在关系中，“一方”使用主关键字或候选索引关键字，而“多方”使用普通索引关键字。在本例中，也就是把“学生”表中的“学号”字段加到“成绩”表中，因为“一”个学生有“多”个成绩。关系型数据库系统使用学号来确定每个成绩记录所属的学生。

多对一关系与一对多关系是相对的，它们唯一不同的就是选择哪一个表作为父表（即关系的基础表，子表是寻找与父表中每个记录相关的记录的表）。

通常，在一对多（多对一）关系中，把“多”表作为这个关系的父表，“一”表作为子表，这样用户可以发现“一”表中有且只有一个记录与“多”表中的记录相对应。如在本例中，把“成绩”表作为父表，这样，在“成绩”表中任意选定一个记录，都能在“学生”表中找到一个学生记录与之对应。

1.3.3 多对多关系

在多对多关系中，表A的一个记录在表B中可以对应多个记录，而表B的一个记录在表A中也可以对应多个记录。这种情况下，在向数据库系统正确指定关系之前，需要改变数据库的设计。

要确定表间的多对多关系，建立多对多关系的数据库，对构成关系的双方进行了解是非常重要的。例如，在教务管理信息系统中学生与课程之间的关系，一个学生可以选择多门课程来学习，同时每门课程又可接纳多个学生。“学生”表中的一个记录对应于“课程”表中的多个记录，同样“课程”表中的一个记录又对应于“学生”表中的多个记录。

在这里，这两个表的主题——学生和课程具有多对多关系，这提出了数据库设计中的

一个新问题。试想一下，如果把这两个表设计为一对多关系，在“学生”表中加入“课程编号”字段来建立两表之间的关系，这时，如果一个学生每次选定多门课程来学习，那么在“学生”表中，每个学生都需要多个课程记录，而对和学生有关的每门课程记录，它都需要重复相同的课程信息，这就不可避免地造成数据的冗余，这说明不正确的数据库表间的关系会导致不准确的结果。如果把“学生”表中的主关键字段“学号”加入到“课程”表中，也会遇到同样的问题。

解决这个问题的方法就是创建第三个表，把多对多的关系分解成两个一对多关系。这第三个表就称作“纽带表”，因为它在两表之间起着纽带的作用。可以把两个表的主关键字都放在这个纽带表中。本例中建立一个“成绩”表，把“学号”字段和“课程编号”放在这个表中，这样就在“学生”表和“课程”表之间建立了多对多关系，如图1-4所示。

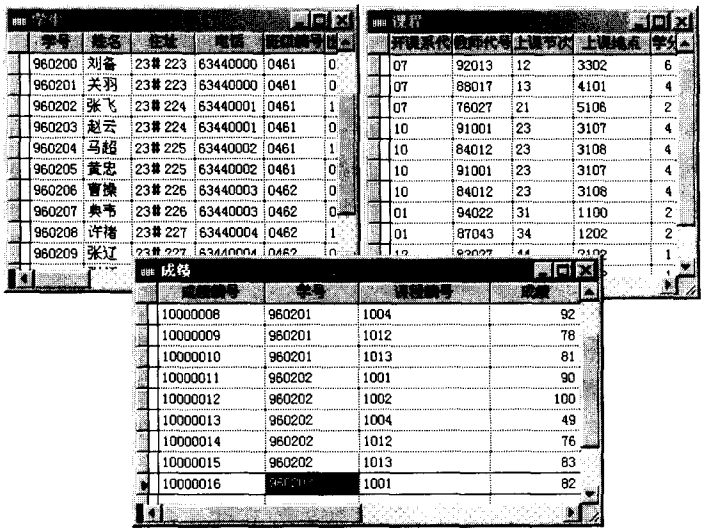


图 1-4　使用“成绩”表为“学生”表和“课程”表建立了多对多关系

纽带表可以只包含它所连接的两个表的主关键字，纽带表也可包含其他信息，如本例中添加了“成绩”字段。

“成绩”表中的每个记录代表了一个成绩记录，我们可以称之为行项。“成绩”表的主关键字由两个字段（来自“学生”表和“课程”表的外部关键字）组成。单独的“学号”字段不能作为这张表的主关键字，因为一个学生可能会和多门课程有关系，这样该字段就不能保持唯一。单独的“课程编号”字段也不能作为主关键字，因为一门课程可能接受许多学生。但在纽带表中，两个字段连在一起就能使每个记录有唯一值。纽带表可以不要自己的主关键字。本例中纽带表有自己的主关键字段“成绩编号”，这是为方便表示不同的“成绩”记录，但并不是必需的。

在上面的数据库中，“学生”表和“课程”表并非直接联系，它们通过“成绩”表间接联系。实际上，“学生”表和“课程”表的多对多关系在数据库中是采用两个一对多关系来代替的：

“学生”表和“成绩”表是一对多关系。每个“学生”可以有多个行项（成绩项），

但每个成绩项只和一个学生有关。

“课程”表和“成绩”表是一对多关系。每门课程可以有许多行项（成绩项）和它联系，但每门成绩只能是一门课程的。

1.4 管理信息系统的基础技术

开发管理信息系统涉及到许多关键技术，其中主要的有数据库技术、计算机网络技术和安全保密技术等。

1.4.1 数据库技术

数据库技术是管理信息系统开发的基础，它的主要功能是以正确的结构和关系存储数据，以备数据的查询、处理和更新。

1. 采用复杂的数据模型

数据模型不仅要描述数据本身的特点，还要描述数据之间的联系。这种联系是通过存取路径来实现的。通过存取路径来表示自然的数据联系，这是数据库与传统文件的根本区别。这样数据不再面向特定的某个或多个应用，而是面向整个应用系统，从而减少了数据冗余，实现了数据共享。

2. 有较高的数据独立性

数据的物理结构和逻辑结构差别可以很大。用户以简单的逻辑结构操作数据而无需考虑数据的物理结构。数据结构分成用户的逻辑结构、整体逻辑结构及物理结构三级。用户的数据和外存中的数据之间的转换由数据库管理系统来实现。为提高效率、减少冗余或增加新的数据，常需改变数据库结构。在改变物理结构时，尽量不要影响整个逻辑结构、用户的逻辑结构以及应用程序，这就是物理数据独立性；在改变整体逻辑结构时，尽量不要影响用户的逻辑结构以及应用程序，这就是整体逻辑数据独立性；在改变用户的逻辑结构时，尽量不要影响整体逻辑结构以及应用程序，这就是逻辑数据独立性。

3. 控制功能

数据库提供下面四个方面的控制功能：

- 数据完整性。保证数据库始终包含正确的数据，系统设计一些完整性检验以确保数据值符合某些规则。
- 数据安全性。保证数据的安全和机密，防止数据丢失和被窃取。
- 数据库的并发控制。避免并发程序之间的干扰，防止数据库被破坏，杜绝提供给用户不正确的数据。
- 存取灵活性。对数据的取存不一定以记录为单位，也可以以字段为单位。这为数据的存取提供了很大的灵活性。

1.4.2 计算机网络技术

计算机网络技术用于MIS，其主要目的是使各子系统之间实现资源共享、互相通信。