

21世纪

高等院校计算机系列教材

数据库管理 与应用开发技术

赵津燕 主 编

李益民 苗 巍 王双立 副主编

138



中国水利水电出版社

www.waterpub.com.cn

21 世纪高等院校计算机系列教材

数据库管理与应用开发技术

赵津燕 主编

李益民 苗巍 王双立 副主编

内 容 提 要

本书较为详细地介绍了数据库应用系统开发所必需的知识。本书分为3篇——绪论、数据库管理系统篇和数据库应用开发技术篇。绪论介绍数据库管理系统和数据库应用开发技术的关系,提出一个数据库应用系统实例——小型教学管理系统,在数据库管理系统篇和数据库应用开发技术篇中,以其为例讲解从设计到形成数据库应用产品的整个开发过程,从而使读者从系统的角度掌握数据库应用的开发技术。数据库管理系统篇以目前市场占有率较高的 SQL Server 2000 为主,讲解如何在数据库服务器端管理和操纵数据库,同时对较为流行但难以入门的 Oracle 数据库管理系统从入门的角度进行介绍。数据库应用开发技术篇以目前最流行的专门的数据库前端开发工具——PowerBuilder 8.0 为主,介绍了客户端应用程序的开发方法,同时还介绍了基于 Web 的数据库应用开发方法,以使读者了解 Web 结构的数据库开发技术。

本书是高等学校计算机专业教学用书,也可作为本专科信息类专业教学用书及数据库短训班的培训教材。

图书在版编目(CIP)数据

数据库管理与应用开发技术 / 赵津燕主编. —北京:中国水利水电出版社, 2004

(21 世纪高等院校计算机系列教材)

ISBN 7-5084-1553-1

I. 数… II. 赵… III. 数据库系统—高等学校—教材 IV. TP311.13

中国版本图书馆 CIP 数据核字 (2004) 第 002987 号

书 名	数据库管理与应用开发技术
作 者	赵津燕 主编 李益民 苗巍 王双立 副主编
出版 发行	中国水利水电出版社 (北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 63202266 (总机)、68331835 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787×1092mm 16 开本 17.25 印张 388 千字
版 次	2004 年 2 月第 1 版 2004 年 2 月第 1 次印刷
印 数	0001—5000 册
定 价	24.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

前 言

随着计算机网络技术的发展,绝大部分的计算机应用成为了基于网络的应用,因而多层体系结构数据库技术得到广泛应用。数据库应用系统的开发,已经由一体的开发分离为服务器端数据库的开发、客户端应用程序的开发以及中间件的设计等部分。FoxBASE、FoxPro 等小型数据库管理系统,已经无法满足发展着的技术的需要,SQL Server、Oracle、Sybase 等大型数据库管理系统迅速取而代之。这些大型数据库管理系统,既支持服务器端数据库的开发,同时又作为数据库服务器,负责完成数据库数据的存储管理、安全管理、并发控制、事务管理、完整性维护、查询优化等工作。PowerBuilder、Delphi、Visual Basic、Visual C++ 等开发工具负责完成客户端应用程序的开发,客户端应用程序负责数据请求、数据表现、菜单和用户界面等功能的实现。SQL Server 是微软公司的数据库服务器产品,SQL Server 2000 是其最新版本,以其易操作及友好的界面,赢得了广大用户的青睐。Oracle 数据库系统是 Oracle 公司开发的关系数据库产品,以其开放性和分布处理能力,获得了较高的市场占有率。PowerBuilder 8.0 是 Sybase 公司推出的开发工具,用于开发多层结构的企业级应用系统,该工具功能全面、性能优异。本书从实用的角度出发,系统地介绍了数据库应用系统的开发方法,有机地将服务器端和客户端的设计结合在一起。

全书分为 3 篇 16 章,第一篇“绪论”为第 1 章,介绍数据库管理系统和数据库应用开发技术的关系,提出一个数据库应用系统实例。第二篇“数据库管理系统篇”为第 2 章~第 10 章,主要介绍了 SQL Server 2000 的使用方法,并利用其实施第 1 章中设计的数据库 TMS,同时还介绍了 Oracle 数据库管理系统。第三篇“数据库应用开发技术篇”为第 11 章~第 16 章,以目前最流行的专门的数据库前端开发工具——PowerBuilder 8.0 为主,介绍了客户端应用程序的开发方法,同时还介绍了基于 Web 的数据库应用开发方法,以使读者了解 Web 结构的数据库开发技术。

本书由赵津燕主编,李益民、苗巍、王双立任副主编,由田永清主审。本书第 1 章~第 5 章、第 7 章由赵津燕编写,第 8 章~第 10 章由苗巍编写,第 11 章~第 14 章由李益民编写,第 6 章、第 15 章、第 16 章由王双立编写。参加本书编写工作的还有闫肃、薛曼玲、赵谢秋等。

在本书的编写过程中,北华大学计算机科学技术学院专业综合实验中心的奚立群、赵谢秋和北华大学 2002 届毕业生张勇、孙海峰等给予了多方面的帮助。在此编者向他们表示衷心的感谢。

由于编者水平有限,书中难免存在错误和不妥之处,恳请读者批评指正。

编 者

2003 年 11 月

目 录

前言

第一篇 绪论

第 1 章 数据库应用系统开发综述	1
1.1 数据库系统结构	1
1.1.1 客户/服务器结构的数据库系统	1
1.1.2 多层结构的数据库系统	2
1.2 数据库应用的开发	3
1.2.1 客户端开发工具与数据库管理系统的关系	3
1.2.2 客户端开发工具的作用	4
1.2.3 数据库管理系统的作用	4
1.2.4 应用开发实例：小型教学管理系统	4
实践	8

第二篇 数据库管理系统篇

第 2 章 SQL Server 2000 数据库的管理	9
2.1 SQL Server 2000 系统概述	9
2.1.1 SQL Server 2000 的安装与配置	9
2.1.2 SQL Server 2000 的工具	13
2.2 SQL Server 2000 中数据库的创建	14
2.2.1 数据库的存储	14
2.2.2 系统数据库	15
2.2.3 SQL Server 的特殊用户	16
2.2.4 用户数据库的创建	17
2.3 SQL Server 2000 中的数据库管理	19
2.3.1 数据库的修改	19
2.3.2 数据库的删除	20
2.3.3 数据库对象的管理	20
实践	26
第 3 章 SQL Server 2000 数据库的安全性和完整性	27
3.1 SQL Server 2000 数据库的安全性	27
3.1.1 数据库的安全性概念	27

3.1.2	验证模式	27
3.1.3	角色管理	28
3.1.4	权限管理	32
3.2	SQL Server 2000 中数据库的完整性	35
3.2.1	数据库的完整性概念	35
3.2.2	数据完整性的实现	36
实践		37
第 4 章	SQL Server 2000 数据库的备份与恢复	38
4.1	SQL Server 2000 中数据库的备份	38
4.1.1	何时备份	38
4.1.2	备份方法	39
4.1.3	备份介质	40
4.1.4	备份需要考虑的其他事项	40
4.1.5	备份用户数据库	41
4.1.6	备份系统数据库	44
4.2	SQL Server 2000 中的数据恢复	44
4.2.1	数据恢复概述	44
4.2.2	验证备份	44
4.2.3	恢复方法	46
实践		48
第 5 章	Transact-SQL 语言	49
5.1	数据类型、变量和运算符	49
5.1.1	数据类型	49
5.1.2	变量	50
5.1.3	运算符	51
5.2	SELECT 语句	51
5.2.1	单表查询	52
5.2.2	连接查询	54
5.2.3	SELECT 语句的一般格式	56
5.3	使用 T-SQL 编程	57
5.3.1	批处理	57
5.3.2	脚本	58
5.3.3	流控制语句	58
5.4	函数	62
5.4.1	系统函数	62
5.4.2	用户自定义函数	65
5.5	事务	69

5.5.1	事务的概念	69
5.5.2	事务的特性	70
5.5.3	管理事务	70
5.5.4	事务控制语句.....	70
5.5.5	分布式事务	71
实践		72
第 6 章	SQL Server 2000 数据库的索引和视图	73
6.1	SQL Server 2000 中数据库的索引.....	73
6.1.1	索引的作用	73
6.1.2	索引的创建	73
6.1.3	索引的管理	76
6.2	SQL Server 2000 中数据库的视图.....	77
6.2.1	视图的作用	77
6.2.2	视图的创建	77
6.2.3	视图的管理	79
实践		82
第 7 章	SQL Server 2000 存储过程和触发器	83
7.1	存储过程概述	83
7.2	存储过程的使用	84
7.2.1	存储过程的创建.....	84
7.2.2	存储过程的修改.....	87
7.2.3	存储过程的删除.....	88
7.2.4	存储过程的执行.....	88
7.2.5	存储过程的设计技巧.....	88
7.3	触发器概述	89
7.4	触发器的使用	89
7.4.1	触发器的工作原理.....	89
7.4.2	触发器的创建.....	90
7.4.3	触发器的修改.....	94
7.4.4	触发器的删除.....	94
7.4.5	触发器的设计技巧.....	95
实践		96
第 8 章	Oracle 数据库系统简介	97
8.1	Oracle 数据库系统概述.....	97
8.1.1	Oracle8 的特点.....	97
8.1.2	Oracle 的网络计算机体系结构	98
8.2	Oracle 8 的安装.....	99

8.2.1 安装前的准备.....	99
8.2.2 Oracle 8 服务器的安装过程.....	100
8.3 配置 Oracle 8.....	102
8.3.1 Oracle 数据库服务器的启动和关闭.....	102
8.3.2 控制服务器的启动和关闭.....	103
8.3.3 连接到 Oracle 8.....	105
8.4 Oracle 的网络连接.....	106
8.4.1 Net8 的功能.....	106
8.4.2 实现一个 Net8 网络.....	106
实践.....	108
第 9 章 执行 SQL 语句的交互环境——SQL*Plus	109
9.1 SQL*Plus.....	109
9.1.1 SQL*Plus 环境.....	109
9.1.2 在 Oracle 中用 SQL 进行基本数据库的访问.....	109
9.2 Oracle 的 SQL.....	110
9.2.1 使用函数.....	110
9.2.2 连接查询.....	111
9.2.3 子查询.....	113
9.2.4 格式化输出.....	116
9.3 过程 SQL 语言——PL/SQL.....	121
实践.....	123
第 10 章 Oracle 数据库的管理	124
10.1 Oracle 数据库管理系统的体系结构.....	124
10.1.1 Oracle 实例.....	124
10.1.2 Oracle 数据库.....	127
10.2 Oracle Enterprise Manager (企业管理器).....	129
10.3 管理数据库.....	130
10.3.1 创建数据库.....	131
10.3.2 管理数据库.....	135
10.4 数据库安全性管理.....	139
10.4.1 管理用户.....	140
10.4.2 管理角色.....	145
10.4.3 管理概要文件.....	148
实践.....	150

第三篇 数据库应用开发技术篇

第 11 章 PowerBuilder 8.0 概述	151
---	------------

11.1	系统要求	151
11.2	安装	151
11.3	操作界面	152
11.4	PowerBuilder 8.0 编程	157
11.4.1	PowerScript 语言简介	157
11.4.2	PowerScript 语言的使用	158
11.4.3	在 PowerBuilder 中使用 SQL 语句	159
11.5	一个简单应用的创建——教学管理系统的创建	161
11.6	窗口	162
11.7	菜单	172
	实践	177
第 12 章	PowerBuilder 8.0 的数据库操作	178
12.1	数据源	178
12.1.1	连接到数据库	178
12.1.2	选择数据	179
12.1.3	定义数据源实例——教学管理系统数据源的创建	182
12.2	数据库接口	183
12.2.1	创建数据库连接对象	183
12.2.2	使用连接对象连接数据	184
12.3	数据窗口操作	188
12.3.1	创建数据窗口	188
12.3.2	使用数据窗口操纵数据	190
12.3.3	数据窗口的动态修改	197
12.3.4	数据窗口应用实例——教学管理系统数据窗口的创建	200
	实践	210
第 13 章	运用高级技术实现更灵活的应用	211
13.1	使用 INI 文件实现数据库连接	211
13.2	在 PowerBuilder 8.0 应用程序中动态使用 SQL 语句	213
13.2.1	动态 SQL 概述	213
13.2.2	动态 SQL 详解	214
	实践	219
第 14 章	可执行应用的建立和应用程序的发布	220
14.1	可执行应用的建立	220
14.2	应用的发行环境	224
14.3	安装盘的制作	227
	实践	228
第 15 章	其他编程语言的数据库应用程序开发	229

15.1 VC++中使用 ODBC 访问数据库	229
15.1.1 与数据库有关的类.....	229
15.1.2 实现 ODBC 数据库的开发过程.....	233
15.2 Delphi 中的数据库程序开发.....	241
15.2.1 常用数据访问部件介绍.....	241
15.2.2 常用数据感知部件介绍.....	246
实践.....	249
第 16 章 Web 数据库应用程序设计	250
16.1 Web 数据库应用程序设计简介	250
16.1.1 Web 数据库应用程序结构	250
16.1.2 HTML 语言初步	251
16.1.3 ASP 简介	254
16.1.4 ADO 简介	258
16.2 基于 Web 的教学管理系统设计	260
实践.....	265
参考文献	266

第一篇 绪论

第1章 数据库应用系统开发综述

数据库应用系统是在数据库管理系统（DBMS）支持下运行的复杂的计算机应用系统。对于不同层次结构的数据库系统来说，开发其应用的方法有所不同。因而要学习数据库应用系统的开发方法，首先应该了解数据库系统的结构。本章将介绍数据库系统的结构，主要介绍客户/服务器结构的数据库系统的有关知识，以及该结构下应用系统开发的基本知识。

1.1 数据库系统结构

数据库系统的体系结构分为单机系统、集中式系统、客户/服务器系统和分布式系统。

单机数据库系统是由同一台计算机完成所有数据库系统的工作，包括存储、处理、管理及使用数据库系统等。

集中式数据库系统中，所有的程序都运行在一台宿主计算机中，所有的数据处理都在宿主计算机中进行，用户通过本地终端或远程终端存取数据库内容。

客户/服务器数据库系统的软件按逻辑功能分为客户端软件和服务器端软件，它们运行在各自的结点上，各负其责，协调工作。

分布式数据库系统由一个概念数据库组成，这个概念数据库的数据存储在网络中多个结点的物理数据库中。

1.1.1 客户/服务器结构的数据库系统

通常所说的客户（client）/服务器（server）结构既可以指硬件的结构，也可以指软件的结构。这里主要指的是后者。软件的客户/服务器结构是指把一个软件系统或应用系统按逻辑功能分成若干个组成部分，例如，用户界面、表示逻辑、事务逻辑、数据访问等。这些软件成分按照其相对角色不同区分为客户软件和服务器软件，客户软件能够请求服务器软件的服务。例如，客户软件负责数据的表示和应用，请求服务器软件为其提供数据的存储和检索服务。客户软件和服务器软件可以分布在网络中不同的计算机结点上，也可以放置在同一台计算机上。

客户/服务器结构的数据库系统中，客户与服务器通过消息传递机制进行对话，客户请求程序首先通过网络协议（如 TCP/IP 及 IPX/SPX 等）与服务器程序进行连接。即由客户端发出请求给服务器端，服务器进行相应的处理，然后送回客户端。客户/服务器消息传递

如图 1.1 所示。

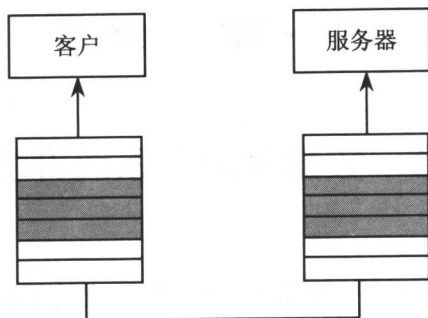


图 1.1 客户/服务器消息传递

在客户/服务器结构数据库系统中，服务器的数目可以是一个，也可以是多个。当系统中存在多个数据库服务器时，就形成了分布式系统。

1. 客户/服务器结构的数据库系统实现技术

客户/服务器结构的数据库系统实现主要依赖以下技术：

(1) 开放的数据库访问接口。目前，市场上有许多数据库产品，客户端开发工具也为数不少。不管选择何种数据库和何种客户端开发工具，都存在客户应用如何访问数据库的问题。开放的数据库访问接口就是解决这个问题。

- 开放数据库互连 ODBC (Open DataBase Connectivity)。ODBC 是 Microsoft 公司提供的一种访问数据库的统一界面标准，使应用程序可以操纵数据库。ODBC 通过其驱动程序提供了数据库的独立性。应用程序可以动态地连接到不同类型数据库的 ODBC 驱动程序，以操纵不同类型的数据库。
- Open Client/Open Server 接口。一些数据库厂商提供了专用的数据库接口，以支持客户软件对数据的访问。包括 Open Client 接口和 Open Server 接口。

(2) 存储过程 (Stored Procedure)。存储过程是用 SQL 语句编写的能够完成一定功能的程序。它有自己的名字，在需要使用它时执行它，即可自动实现该存储过程所定义的功能。

(3) 分布数据管理。客户/服务器结构的数据库系统中，如果数据分布在多个数据库服务器上，则需要采用分布数据管理技术。

2. 客户/服务器结构的数据库系统的优点

客户/服务器结构的数据库系统能够提供数据和服务的无缝集成，为联机事务处理提供高性能，同时提供了开放的系统结构，提高了应用开发生产率。

1.1.2 多层结构的数据库系统

以 Web 服务器为中心，采用 TCP/IP、HTTP 为传输协议，客户端通过浏览器访问 Web 服务器以及与 Web 相连的后台数据库，称为 BWD (Browser Web Database) 模式，如图 1.2 所示。BWD 模式突破了传统的客户/服务器模式，是一种多层次的客户/服务器

模式。它有一个明显的部分就是中间的代理部分,代理的目的是集成后端不同的服务器,使它们的资料及操作界面都能以统一的方式呈现。它的优点是,用户不管使用何种浏览器来访问 Internet 上的 Web 站点,都可以看到相同的操作界面,符合企业对于统一操作界面的要求,而且保留了整体的弹性,后端的服务器仍然可以随时根据需要进行补充和修改。

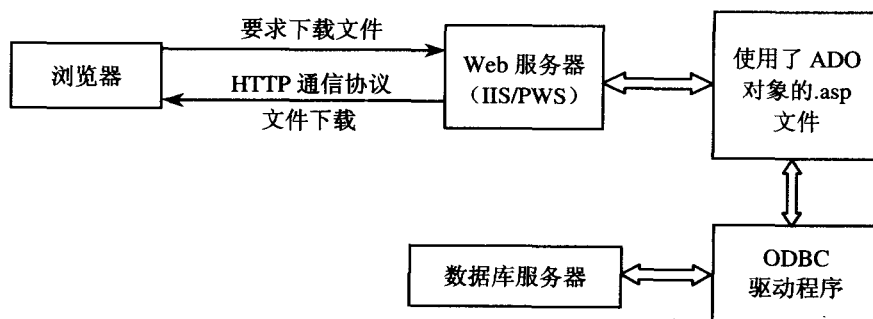


图 1.2 BWD 模式

1.2 数据库应用的开发

了解了数据库系统结构以后,对数据库应用开发会有初步的认识。但作为一个软件设计人员如何能真正掌握数据库应用系统的开发技术,还要进一步学习客户端开发工具的使用与数据库管理系统 (DBMS) 的知识。

1.2.1 客户端开发工具与数据库管理系统的关系

使用客户端开发工具开发客户应用程序,这个程序首先建立一个与数据库的通信渠道,然后将用户的需求以某种方式传送给数据库服务器。在应用程序接收到数据库服务器返回的数据后,分析并呈现给用户。因此,客户应用程序只完成请求和表现数据的工作,是用户操作计算机的人机界面,大多数数据处理是由服务器完成的。

数据库服务器是一个存取数据和管理数据的软件,它针对客户的请求为客户提供数据服务。这些服务包括数据插入、修改和查询等。客户对数据库服务器提出请求用的语言是 SQL (Structured Query Language)。

因此,客户端开发工具与数据库是“配合”与“协调”的关系。客户应用程序完成数据请求、数据表现、菜单、界面等表现方面的工作,属于与用户接触最近的部分,因而开发客户应用程序的工具又称为前端开发工具;而数据库管理系统完成数据库的存储管理、安全管理、并发控制、事务管理、完整性维护、查询优化等工作,它离用户相对较远,其所用开发工具因而又称为后端开发工具。图 1.3 表示了客户应用程序与数据库配合的工作方式。

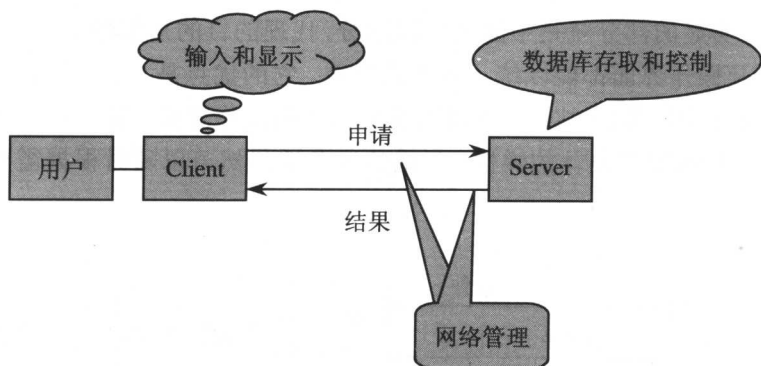


图 1.3 客户应用程序与数据库配合的工作方式

1.2.2 客户端开发工具的作用

客户端开发工具一般用来设计窗口和定义窗口中的控制；生成菜单；嵌入 SQL 语句操纵数据库；极方便地生成和维护数据库；编写应用服务器程序。

目前市场上的数据库应用前端开发工具有很多，如 Visual C++、Visual Basic、Delphi 等，但它们都不是专用的，对于复杂的数据库应用，还是选择专用工具为好。本书后面主要介绍的 PowerBuilder 就是最流行的专用数据库应用开发工具之一，它也是目前市场占有率最高的专用数据库应用开发工具。

1.2.3 数据库管理系统的作用

数据库是应用系统的核心，它的作用主要有以下几个方面：接受和执行 SQL 语句；事务处理；并发控制；查询优化；数据安全性保护；完整性约束；支持存储过程和触发器；数据备份与恢复。

实际上，从系统组成来看，一个完整的数据库应用一般由数据库、数据库管理系统（及其开发工具）、应用系统、数据库管理员和用户构成。数据库是应用系统的核心，选择合适的数据库管理系统是应用系统开发成功与否的关键问题。目前市场上应用较多的数据库产品有很多，如 Oracle、Sybase、MS SQL Server 等，其中 MS SQL Server 的最新版本 SQL Server 2000，具有操作简单、技术先进、功能完善的特点，不失为一个好的选择。

应当指出的是，数据库的建立、使用和维护等工作只靠一个 DBMS 远远不够，还要有专门的人员来完成，这些人称为数据库管理员（database administrator，简称 DBA）。

综上所述，无论是作为数据库管理员还是应用的开发者都应该熟悉 DBMS，因此掌握一种数据库管理系统（如 MS SQL Server）对于一个软件开发人员就像掌握一门程序设计工具一样重要。也就是说，要完成数据库应用系统的开发，不仅要掌握客户端的开发工具，还要掌握服务器端的数据库管理系统。

1.2.4 应用开发实例：小型教学管理系统

前面介绍了数据库应用开发工具及其作用，现在以一个小型教学管理系统为例说明数

数据库开发过程，并在其后的章节中进行详细的介绍。

按照规范化设计的方法，考虑数据库及其应用系统开发的全过程，将数据库设计分为6个阶段：需求分析、概念结构设计、逻辑结构设计、物理结构设计、数据库实施、数据库运行与维护。下面将简单描述整个过程，详细的实现将在后续的内容中阐述。

1. 该项目的基本需求分析

假设我们正在为某学校设计一个小型教学管理系统，需要完成以下功能：

- 学生基本情况管理。学生基本情况表由所在系号、学号、姓名、性别和出生时间等组成。对该表的管理包括数据录入、修改、查询、删除等操作。
- 课程表管理。课程表由课程号、课程名称、直接前序课号和学分组成。对该表的管理包括数据录入、修改、查询、删除等操作。
- 学生选课表管理。学生选课表由所在系号、学号、课程号和成绩组成。对该表的管理包括数据录入、修改、查询、删除等操作。
- 成绩管理。该管理包括成绩录入、修改、查询、删除、统计等操作。其中查询操作包括按班级查询成绩单、按个人查询成绩单、按课程查询不及格学生名单等查询操作，统计操作包括某门课的成绩统计、个人成绩和学分等统计操作。
- 系统维护。每到学期开始时，我们需要对全校各班人数进行核对，学生毕业3年后，其信息将从数据库中导出到软盘上作为历史数据保存，而数据库中的相应信息将被删除。为满足院、系、专业、班级的变化需要，应有对相应的代码表进行维护的功能。

2. 数据库的概念设计

将需求转换为数据库的概念模型（又称为 data schema），这一步同具体的数据库产品没有关系。可以采用 E-R 图来描述概念模型。

E-R 图主要由3个组件组成：

(1) 实体 (entity)：客观存在并可相互区分的事物。如：一个教师、一门课程。在 E-R 图中一般用长方形表示。它在 DB 概念设计中被转化为表。

(2) 关系 (relation)：独立的实体相互之间的关系。如：教师和课程的关系是讲授关系。在 E-R 图中一般用菱形表示。在 DB 的概念设计中，关系是通过主键和外键来描述的，用于维护参照完整性。实体间的关系虽然复杂，但抽象以后，可把它们归结为3类：

- 一对一联系 (1:1)：对于实体集 A 中的每一个实体，实体集 B 中至多有一个实体与之联系，反之亦然。把这种关系定义为 1:1。如：一个班级有一个班主任，每个班主任只负责一个班。那么，班主任和班级之间的关系是一对一关系。
- 一对多联系 (1:n)：对于实体集 A 中的每一个实体，实体集 B 中有 n ($n \geq 0$) 个实体与之联系，反之，对于实体集 B 中的每一个实体，实体集 A 中至多有一个实体与之联系。把这种关系定义为 1:n。如：一个班级有多个学生，而每个学生只在一个班中。那么，学生和班级之间的关系是一对多的关系。
- 多对多联系 (m:n)：对于实体集 A 中的每一个实体，实体集 B 中有 n ($n \geq 0$) 个实体与之联系，反之，对于实体集 B 中的每一个实体，实体集 A 中也有 m ($m \geq 0$)

个实体与之联系。把这种关系定义为 $m:n$ 。如：一个学生可以选择多门课，而每门课可以有多个学生选择。那么，学生和课程之间的关系是多对多的关系。

(3) 属性 (attribute)：实体所具有的某一特征。如：学生姓名、性别、出生日期等。在 E-R 图中一般用椭圆形表示。在 DB 的逻辑设计中，属性被转化为表中的列或字段。

下面举个例子，见图 1.4。通过 E-R 图描述了两个实体学生 (student) 和课程 (course) 的关系。学生有 5 个属性：系号 (dno)、学号 (sno)、姓名 (sname)、性别 (sssex) 和出生时间 (sbirth)。课程有 4 个属性：课程号 (cno)、课程名称 (cname)、前序课号 (spno) 和学分 (ccredit)。

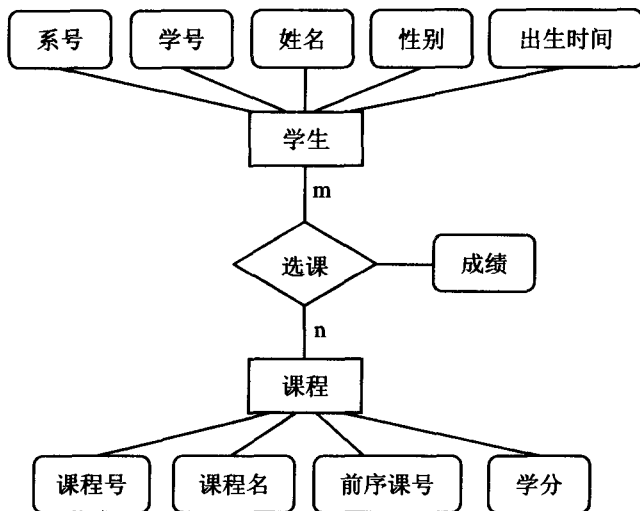


图 1.4 E-R 图例

按照上述方法，读者可以完成教学管理系统的所有 E-R 模型。

3. 数据库的逻辑设计

可以采用两种方法将 E-R 模型转化为逻辑数据库。一种方法是采用如 PowerDesign 那样的系统分析与设计工具，另一种方法是用手工的方法将它们转化。下面描述手工的转化方法，它仍遵循工具转化的原理。

在 E-R 图中，实体转换为表名，属性转换为列。E-R 图中的关系是通过主键/外键的参照关系体现的。针对 3 种关系，应该遵循以下原则：

- 表 A 和表 B 是一对一联系 (1:1)：也就是说，既可以将表 A 的主键 (记为 Pa) 列添加到表 B 中充当外键，也可以将表 B 的主键 (记为 Pb) 列添加到表 A 中充当外键。
- 表 A 和表 B 是一对多联系 (1:N)：必须把表 A 的主键 (记为 P1, 1 的意思是指 parent 表) 列添加到表 B 充当外键。如：在教学管理系统中，department 表和 student 表是一对多的关系，所以在 student 表中 dno 列参照 department 表中的主键 dno。
- 表 A 和表 B 是多对多联系 (M:N)：除了生成表 A 和表 B 外，还应该生成一张关系表。这个关系表的列是：由表 A 的主键+表 B 的主键+关系自己的属性。例

如：student 表和 course 表是多对多的关系。那么，除了生成 student 表和 course 表外，还应该生成一张关系表 sc。

最后，生成了如图 1.5 所示的教学管理系统的逻辑数据库。

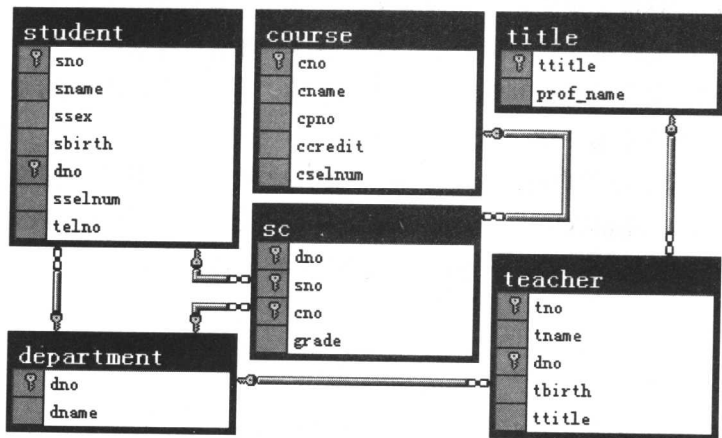


图 1.5 tms 数据库的数据关系图

4. 数据库的物理设计

在完成数据库的逻辑设计后，就要做数据库的物理设计了。物理设计首先是根据逻辑设计的数据库的结构和以后的数据量、查询和更新的频率，来决定选用哪个 DBMS，如 Microsoft SQL Server、Oracle、IBM DB2、Sybase 等。这里假定选择 Microsoft SQL Server。

5. 数据库实施

根据具体的 DBMS 来完成数据库的实施。在 Microsoft SQL Server 中，需要创建数据库和表等其他数据库对象（参见第 2 章）。创建的方法是采用数据定义语言（DDL）或图化工具来物理实现数据库模式。

在创建了数据库之后，需要考虑数据的完整性、数据的安全性等要求。数据的完整性是指数据的正确性、有效性和相容性。系统提供必要的功能，保证数据库中的数据在输入、修改过程中符合原来的定义和规定，如：性别是‘男’和‘女’之中的一个，课程号是惟一的等。在 SQL Server 中，通过默认和规则编写触发器、存储过程来实现数据完整性要求（参见第 7 章）。值得注意的是，完整性的实现可以在数据库设计的任一阶段及数据库的各个级别上实现。在创建表时，应该考虑以下表约束：

- 表名惟一：由 SQL Server 实施。
- 列名惟一：由 SQL Server 实施。
- 行惟一：由主键来定义。

在 SQL Server 中，系统没有实施列约束，但数据库设计者应该考虑列约束，这就需要自行设计并实现。

- 非空（Not Null）：要求列中有输入项。
- 无重复（Not Duplicate）：要求列中无重复值（可通过创建：主键、惟一约束、