

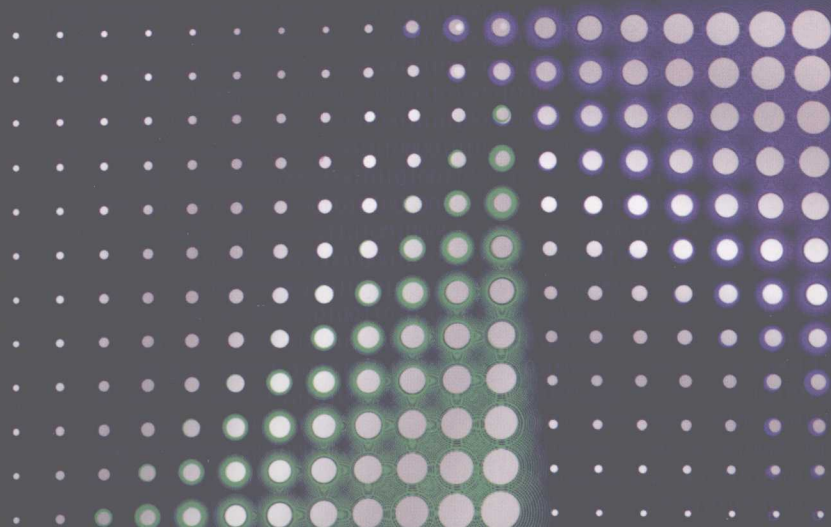


新编计算机类本科规划教材

网络数据库 实用教程

——基于Visual Studio 2005 和 SQL Server 2005

芦丽萍 柳彩志 编著 毕广吉 审校



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

基于 Visual Studio 2005 和 SQL Server 2005

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书从数据库的概念入手,有针对性地介绍了数据库管理方面的知识,并结合丰富的上机实例练习,循序渐进地指导读者掌握 SQL Server 2005。在充分理解并熟练掌握数据库的基础上,本书介绍了当前软件开发领域广泛使用的 ASP.NET 2.0 技术,并具体介绍了 Visual Studio 2005 集成开发工具。作为一个综合应用的实例,本书详细介绍了基于 Visual Studio 2005 和 SQL Server 2005 环境开发的“高校学生管理系统”的全过程。

本书通俗易懂、重点突出、实例丰富,具有较强的实用价值,适合作为高校计算机专业或相关专业“数据库技术及网络应用”课程的教材,也可供其他学习及开发网络数据库的读者参考。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

网络数据库实用教程:基于 Visual Studio 2005 和 SQL Server 2005 / 芦丽萍, 柳彩志编著.

—北京:电子工业出版社,2008.6

新编计算机类本科规划教材

ISBN 978-7-121-06799-0

I. 网… II. ①芦…②柳… III. 关系数据库—数据库管理系统, SQL Server 2005—高等学校—教材 IV. TP311.138

中国版本图书馆 CIP 数据核字(2008)第 076687 号

责任编辑:冯小贝

印 刷:北京市李史山胶印厂
装 订:

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:19.25 字数:493 千字

印 次:2008 年 6 月第 1 次印刷

定 价:29.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

前 言

随着 Internet 的普及和推广, 数据库技术得到了迅猛的发展。数据库是一门应用很广、实用性很强的技术, 现已广泛应用到了社会生活的各个领域。为了能够适应教学和初学者对网络数据库编程学习和参考的需求, 作者根据多年的教学经验和体会精心编写了本教程。

本书使用最新的可视化编程环境 Visual Studio 2005 在最新的数据库平台 SQL Server 2005 上介绍网络数据库编程技术, 书中的内容代表了数据库程序设计和应用的先进水平。本书通过丰富的实例和上机实践, 使读者加深对数据库基础知识和基本原理的理解, 逐步掌握 ASP.NET 的应用开发方法和技巧, 达到开发实际网络数据库应用系统的目的。

本书是一本网络数据库编程的入门教材, 具有以下几个优点。

- **学习起点低, 知识层次深。**本书的读者定位是网络数据库技术的初学者, 可以是零起步, 但本书的内容全面, 知识层次深。不仅系统讲解了当前流行的 SQL Server 数据库的基础知识和高级操作, 而且全面介绍了 .NET 最先进的 Web 应用程序开发技术。
- **内容实用、编排合理、实践性强。**本书的第 1 章~第 8 章系统讲解了数据库技术, 第 9 章~第 11 章讲解了 ASP.NET 技术, 第 12 章给出了一个综合实例, 是第 1 章~第 11 章内容的综合应用。全书的安排注重实践性, 实例丰富, 易于上手。
- **语言简练、通俗易懂、容易入门。**本书的作者充分考虑到初学者的接受能力和学习特点, 将晦涩难懂的高深理论用通俗易懂的语言表达出来, 使读者轻松入门。
- **习题类型丰富, 便于巩固练习。**本书每章后面均附习题, 题型丰富, 并且在附录中提供了部分习题的参考答案。

本书通俗易懂、重点突出、实例丰富, 具有较强的实用价值, 适合作为高校计算机专业或相关专业“数据库技术及网络应用”课程的教材, 也可供其他学习及开发网络数据库的读者参考。

本书第 1 章~第 8 章和附录 B 由芦丽萍编写, 第 9 章~第 11 章由柳彩志编写, 第 12 章和附录 A 由王雪编写。本书最后由毕广吉教授进行审校, 教学课件由赵晶晶制作完成, 在此表示感谢。

本书免费提供电子教案及“高校学生管理系统”源程序代码, 教师可到华信教育资源网 (www.huaxin.edu.cn) 下载。

由于作者水平有限, 书中的疏漏或错误之处在所难免, 恳请广大读者批评指正。

目 录

第 1 章 数据库概述	(1)
1.1 数据管理技术的发展	(1)
1.1.1 人工管理阶段	(1)
1.1.2 文件系统阶段	(1)
1.1.3 数据库系统阶段	(2)
1.2 数据模型	(2)
1.2.1 概念模型	(3)
1.2.2 实施模型	(4)
1.3 数据库系统	(6)
1.3.1 数据库系统的组成	(6)
1.3.2 数据库管理系统	(7)
1.3.3 数据库系统的三级模式结构	(7)
1.4 关系数据库	(9)
1.4.1 关系数据模型	(9)
1.4.2 关系代数运算	(10)
1.4.3 关系代数与数据查询	(15)
1.4.4 数据冗余和数据完整性	(16)
1.4.5 常用关系数据库简介	(17)
小结	(19)
习题 1	(19)
第 2 章 SQL Server 2005 的安装与管理	(22)
2.1 SQL Server 2005 的安装	(22)
2.1.1 SQL Server 2005 的主要版本	(22)
2.1.2 安装 SQL Server 2005 系统的软硬件需求	(23)
2.1.3 安装 SQL Server 2005 的具体步骤	(25)
2.2 SQL Server 2005 基础知识	(28)
2.2.1 SQL Server 2005 的新特性	(28)
2.2.2 系统数据库和示例数据库	(29)
2.2.3 SQL Server 2005 的基本使用方法	(30)
2.3 SQL Server 2005 安全管理机制	(31)
2.3.1 客户操作系统的安全性	(31)
2.3.2 SQL Server 服务器验证模式	(31)

2.3.3 数据库的安全性	(32)
2.3.4 数据库对象的安全性	(38)
小结	(38)
习题 2	(38)
第 3 章 使用 SQL Server Management Studio 操纵数据库和表	(40)
3.1 数据库管理	(40)
3.1.1 与数据库相关的几个基本概念	(40)
3.1.2 数据库的创建	(40)
3.1.3 数据库的修改	(42)
3.1.4 数据库的删除	(43)
3.1.5 数据库的分离与附加	(44)
3.1.6 数据库的备份与还原	(45)
3.2 数据库表管理	(48)
3.2.1 与数据库表相关的几个基本概念	(48)
3.2.2 数据库表结构的创建	(49)
3.2.3 数据库表结构的修改	(51)
3.2.4 数据库表的删除	(51)
3.2.5 建立数据库表之间的联系	(51)
3.2.6 数据库表记录的增、删、改	(53)
3.3 创建数据库实例——“学生成绩管理”数据库	(54)
3.3.1 创建数据库	(54)
3.3.2 创建数据库表结构	(56)
3.3.3 创建数据库关系图	(57)
3.3.4 输入记录	(58)
小结	(60)
习题 3	(60)
第 4 章 T-SQL 程序设计基础	(62)
4.1 SQL 语言和 T-SQL 语言	(62)
4.1.1 SQL 语言简介	(62)
4.1.2 T-SQL 语言简介	(62)
4.1.3 T-SQL 语句的功能及语法规则	(62)
4.2 T-SQL 的标识符	(63)
4.2.1 标识符类型	(63)
4.2.2 标识符命名规则	(64)
4.2.3 特殊意义的标识符	(64)
4.3 T-SQL 的数据类型	(64)
4.3.1 整型数据类型	(64)
4.3.2 货币数据类型	(65)

4.3.3	精确小数数据类型	(65)
4.3.4	近似小数数据类型	(66)
4.3.5	日期和时间数据类型	(67)
4.3.6	字符串数据类型	(67)
4.3.7	Unicode 字符串数据类型	(68)
4.3.8	二进制字符串数据类型	(68)
4.4	T-SQL 的运算符及表达式	(69)
4.4.1	算术运算符及表达式	(69)
4.4.2	赋值运算符及表达式	(69)
4.4.3	字符串运算符及表达式	(70)
4.4.4	关系运算符及表达式	(70)
4.4.5	逻辑运算符及表达式	(71)
4.4.6	位运算符及表达式	(72)
4.4.7	各种运算符的优先级别	(73)
4.5	T-SQL 中的变量	(74)
4.5.1	局部变量	(74)
4.5.2	实例: 局部变量的应用	(75)
4.5.3	全局变量	(77)
4.5.4	实例: 全局变量的应用	(77)
4.6	T-SQL 中的语句	(79)
4.6.1	注释语句	(79)
4.6.2	批处理及结束标志 go	(79)
4.6.3	begin...end 语句块	(80)
4.6.4	条件语句	(81)
4.6.5	循环语句	(84)
4.6.6	return 语句	(85)
4.7	T-SQL 函数	(86)
4.7.1	数学函数	(86)
4.7.2	字符串函数	(88)
4.7.3	日期和时间函数	(90)
4.7.4	系统函数	(91)
4.7.5	类型转换函数	(91)
4.7.6	用户自定义函数	(92)
小结		(94)
习题 4		(94)
第 5 章	用代码操纵数据库和表	(96)
5.1	T-SQL 数据定义及基本操作	(96)
5.1.1	T-SQL 语句的输入和运行环境	(96)

5.1.2	实例：一个简单的 T-SQL 程序	(98)
5.1.3	数据库的创建与删除	(98)
5.1.4	实例：用代码创建数据库	(100)
5.1.5	表结构的创建、修改与删除	(101)
5.1.6	实例：用代码创建数据库表	(102)
5.1.7	插入数据行	(103)
5.1.8	更新数据	(104)
5.1.9	删除数据	(104)
5.1.10	实例：用代码操纵数据库表	(104)
5.2	T-SQL 简单查询	(107)
5.2.1	select 语句的基本用法	(107)
5.2.2	使用 where 子句	(107)
5.2.3	条件表达式中的运算符	(108)
5.2.4	用于特殊查询的关键字	(109)
5.2.5	实例：简单查询荟萃	(110)
5.3	T-SQL 高级查询	(111)
5.3.1	多表查询	(111)
5.3.2	排序查询	(112)
5.3.3	分组查询	(112)
5.3.4	统计函数查询	(113)
5.3.5	嵌套查询	(114)
5.3.6	实例：高级查询荟萃	(114)
	小结	(116)
	习题 5	(116)
第 6 章	视图和索引	(119)
6.1	视图	(119)
6.1.1	视图的概念与意义	(119)
6.1.2	视图的创建与显示	(120)
6.1.3	视图的删除	(125)
6.1.4	修改视图	(127)
6.1.5	视图安全	(129)
6.2	索引	(130)
6.2.1	索引的概念	(130)
6.2.2	索引的特点	(131)
6.2.3	索引的分类	(131)
6.2.4	索引的创建	(132)
6.2.5	索引的删除	(134)
	小结	(136)
	习题 6	(136)

第 7 章 SQL Server 高级应用	(138)
7.1 游标	(138)
7.1.1 游标概述	(138)
7.1.2 声明游标	(138)
7.1.3 打开游标	(139)
7.1.4 读取游标数据	(139)
7.1.5 实例：定义一个滚动游标	(139)
7.1.6 实例：读取游标中的数据并存入变量中	(140)
7.1.7 利用游标更新数据和删除数据	(141)
7.1.8 实例：利用游标统计数据行数	(141)
7.1.9 关闭游标和释放游标	(142)
7.1.10 实例：利用游标删除表中的一行	(143)
7.2 存储过程	(144)
7.2.1 存储过程概述	(144)
7.2.2 存储过程的创建与执行	(144)
7.2.3 实例：不带参数的存储过程	(144)
7.2.4 实例：带输入参数的存储过程	(145)
7.2.5 实例：带输入和输出参数的存储过程	(146)
7.2.6 存储过程的修改、重命名和删除	(148)
7.2.7 实例：修改存储过程	(148)
7.2.8 实例：重命名和删除存储过程	(149)
7.3 触发器	(150)
7.3.1 触发器概述	(150)
7.3.2 触发器的工作原理	(151)
7.3.3 触发器的创建与执行	(151)
7.3.4 实例：显示功能触发器	(151)
7.3.5 触发器的修改、删除和显示	(152)
7.3.6 实例：插入功能触发器	(153)
7.3.7 实例：更新功能触发器	(154)
7.4 事务处理	(155)
7.4.1 事务概述	(155)
7.4.2 事务属性	(156)
7.4.3 事务控制语句	(156)
7.4.4 实例：事务编程	(156)
7.4.5 事务并发处理与并发控制	(157)
小结	(158)
习题 7	(158)

第 8 章 数据库设计	(160)
8.1 数据库设计概述	(160)
8.1.1 数据库与信息系统	(160)
8.1.2 数据库设计的特点	(160)
8.1.3 数据库设计方法和步骤	(161)
8.2 数据库设计过程	(162)
8.2.1 需求分析	(162)
8.2.2 概念结构设计	(163)
8.2.3 逻辑结构设计	(164)
8.2.4 物理结构设计	(165)
8.2.5 数据库实施	(165)
8.2.6 数据库运行和维护	(165)
8.3 关系数据库规范化	(165)
8.3.1 范式	(166)
8.3.2 数据库设计应注意的问题	(166)
8.4 数据库设计实例——培训部管理系统数据库设计	(167)
8.4.1 需求背景分析	(167)
8.4.2 功能模块设计	(168)
8.4.3 数据库设计	(168)
小结	(171)
习题 8	(171)
第 9 章 VB.NET 程序设计基础	(173)
9.1 ASP.NET 与 VB.NET	(173)
9.2 数据类型、变量和表达式	(173)
9.2.1 VB.NET 的数据类型	(173)
9.2.2 VB.NET 的常量和变量	(174)
9.2.3 VB.NET 运算符	(178)
9.3 VB.NET 程序控制语句	(180)
9.3.1 条件语句	(180)
9.3.2 循环语句	(183)
9.3.3 注释语句	(187)
9.4 VB.NET 的子程序和函数	(187)
9.4.1 子程序	(187)
9.4.2 Function 函数	(189)
9.4.3 过程参数的传递方式	(191)
9.4.4 VB.NET 系统内置函数	(191)
习题 9	(194)

第 10 章 ASP.NET 技术及应用	(196)
10.1 ASP.NET 概述	(196)
10.1.1 .NET 框架	(196)
10.1.2 ASP.NET 简介	(196)
10.1.3 ASP.NET 运行环境	(197)
10.2 服务器控件	(198)
10.2.1 HTML 服务器控件	(199)
10.2.2 Web 服务器控件	(203)
10.2.3 验证控件	(211)
10.3 ASP.NET 内置对象	(214)
10.3.1 Page 对象	(214)
10.3.2 Request 对象	(216)
10.3.3 Response 对象	(217)
10.3.4 HttpCookie 对象	(218)
10.3.5 Session 对象	(219)
10.3.6 Application 对象	(221)
10.3.7 Server 对象	(222)
10.3.8 Mail 对象	(224)
习题 10	(226)
第 11 章 ADO.NET 与数据库访问	(228)
11.1 ADO.NET 简介	(228)
11.1.1 ADO.NET 数据库驱动程序	(228)
11.1.2 本章使用的数据库	(229)
11.1.3 使用 Connection 对象连接数据库	(230)
11.2 使用 ADO.NET 内置对象访问数据库	(232)
11.2.1 使用 Command 执行数据库命令	(232)
11.2.2 使用 DataReader 读取数据	(236)
11.2.3 使用 DataAdapter 读写数据	(238)
11.2.4 使用 DataSet 操作数据	(241)
习题 11	(245)
第 12 章 开发高校学生管理系统	(248)
12.1 需求分析	(248)
12.2 系统功能模块设计	(248)
12.3 数据库设计	(249)
12.3.1 设计表结构	(249)
12.3.2 设计表关系	(251)
12.3.3 设计数据库脚本	(251)
12.4 Visual Studio 2005 集成开发环境介绍	(254)

12.4.1	控制浮动面板	(255)
12.4.2	工具箱面板	(255)
12.4.3	解决方案资源管理器	(256)
12.4.4	编辑器和设计器	(256)
12.4.5	属性面板	(256)
12.5	工程框架设计	(257)
12.5.1	系统架构设计	(257)
12.5.2	创建工程	(257)
12.5.3	母版页 (Master Page) 与主题 (Theme)	(259)
12.5.4	系统工程文件一览	(260)
12.6	系统功能模块的设计与实现	(261)
12.6.1	系统用户管理模块	(261)
12.6.2	班级管理模块	(267)
12.6.3	学生基本信息管理模块	(269)
12.6.4	学生成绩管理模块	(270)
12.6.5	学生选课管理模块	(271)
12.6.6	学生专业调整管理模块	(275)
12.6.7	学生奖惩管理模块	(276)
小结		(278)
附录 A	高校学生管理系统配置指南	(279)
附录 B	习题参考答案	(280)
参考文献		(293)

第1章 数据库概述

本章提要

本章将介绍数据库的基本概念,其中包括数据管理技术的发展过程、数据模型的分类、数据库系统的体系结构及关系数据库。在关系数据库中重点讨论关系数据库的基本概念、关系代数运算、数据冗余和数据完整性等概念。

1.1 数据管理技术的发展

数据处理的核心问题是对数据的管理。数据(data)是指所有能通过编码被输入到计算机中并能被程序处理的符号集合,可包含文本、数值、声音、图形、图像、动画等信息。数据管理是指对数据进行组织、分类、存储、检索和维护等操作。数据管理经历了人工管理、文件系统和数据库系统三个发展阶段。

1.1.1 人工管理阶段

20世纪50年代,计算机的发展还处于初级阶段。计算机在硬件方面的特点是运算速度慢、内存小,外存只有磁带、卡片和纸带而没有直接存取的存储设备;在软件方面,只有汇编语言而没有操作系统和数据管理软件,数据处理方式为批处理;当时计算机的应用领域主要是科学计算。在人工管理阶段,数据管理技术具有以下特点。

- **数据不保存。**限于硬件所具备的条件和当时计算机的应用状况,数据不便保存。当研究某一课题时,只能临时将数据输入、计算、输出。
- **无管理数据的软件。**数据完全依赖于程序,由于没有专门的软件对数据进行管理,因此数据由应用程序自己管理。程序员在设计应用程序时,不仅要设计程序的逻辑结构,而且还要设计程序的物理结构。这两种结构只要有一个改变,就要修改相应的程序,系统维护的工作量较大。
- **数据不能共享。**数据是面向应用的,数据和程序不能独立。在两个不同的应用程序中,即使某些数据相同,数据之间也不能共享。因此,各个程序之间会产生大量的数据冗余。

1.1.2 文件系统阶段

20世纪50年代中期到20世纪60年代后期,计算机的发展到了一个新的阶段。在硬件方面,已有了直接存取的存储设备,如磁盘、磁鼓等;在软件方面,出现了高级语言和操作系统;数据处理方式不仅有批处理还有实时联机处理;计算机的应用领域主要包含科学计算和数据管理。在文件系统阶段,数据管理技术具有以下特点:

- **数据可长期保存。**由于有了直接存取设备,进行数据处理时可大量存取数据,并可对数据进行增、删、改、查询等操作,数据以文件形式保存。

- **有管理数据的软件。**操作系统中包含了专门的数据管理软件，称之为文件系统。程序和数据文件的存取由文件系统自动完成，文件系统负责对所有程序和数据文件进行管理。程序员通过文件名存取数据，文件系统自动将程序的逻辑结构转换为物理结构。因此，在文件系统阶段维护程序的工作量大大减少。
- **文件形式多样，数据结构化存储。**文件的形式包含索引文件、链接文件和直接存取文件等，文件的存取基本上以记录为单位，而记录是具有一定结构的。
- **数据冗余度大，共享性低。**在文件系统阶段，数据仍然是面向应用的。相同的数据可能会在多个不同的文件中出现，这就造成了大量的数据冗余。这个时期，数据的共享性差，数据和程序仍然不能独立，不便于进行事务处理。

1.1.3 数据库系统阶段

从 20 世纪 60 年代后期开始，计算机的应用领域越来越广泛，需要计算机管理和处理的数据规模越来越大。在硬件方面，由于大规模集成电路的出现，使得计算机的运算速度加快，内存空间扩大，外存也有了大容量的磁盘、磁鼓、光盘等直接存取设备。然而，在软件方面却出现了价格上升、成本上涨、维护代价过高的问题；数据的实时联机处理也呈大步增长之势，共享数据的需求也越来越大，因而亟需提升数据管理技术的水平。此时，数据库系统应运而生，数据库系统解决了多用户共享数据的问题。在数据库系统阶段，数据管理技术具有以下特点：

- **数据结构化。**在数据库系统中，使用数据模型来描述数据。数据的描述不仅包含数据本身，还包含数据之间的联系。数据是结构化的，这是数据库系统的重要特征，也是实现数据共享的前提条件。
- **数据冗余度小、数据共享、较高的数据独立性。**数据是面向整个系统的，这样可以大大减小数据冗余，节省存储空间，减少数据存取时间，同时也可以避免数据异常。在数据库系统中，能够实现数据共享，支持多个用户的数据存储。数据不依赖于程序，具有较高的独立性。
- **可对数据进行整体控制。**在数据库系统中，多个用户可同时使用数据库资源。为了保证数据库中的数据不被破坏，数据库系统应对数据资源进行整体控制。数据库系统能够提供以下的控制功能：
 - **操作并发控制。**当多个用户同时使用数据库中的数据时，可能会造成数据的不一致性，甚至会出现“死锁”。因此，必须要有控制并发操作的机制。
 - **数据安全性控制。**为了避免数据被非法用户使用或破坏，必须采用一些安全机制。例如定义口令、设置访问权限等，以保护数据的安全。
 - **数据完整性控制。**数据的完整性控制是指保证数据的正确性、有效性和相容性，防止错误的数据被输入到数据库中。

1.2 数据模型

数据模型用图的形式来表达对现实世界中复杂对象的抽象描述。在数据库领域中，用

数据模型描述数据的结构、数据的性质、数据之间的联系、完整约束条件及某些数据的变换规则。根据数据模型的应用目的不同,将数据模型分为两个级别:概念模型和实施模型。

1.2.1 概念模型

概念模型是现实世界的第一层抽象,它把现实世界中的客观对象抽象成一种信息结构,这种信息结构并不依赖于具体的计算机系统和数据库管理软件。现实世界反映到人的大脑中,人们经过认识、选择、命名、分类和组织等过程将其抽象后形成概念模型,并进入到信息世界;然后,将概念模型转换为数据模型,从而走进机器世界。概念模型是现实世界到机器世界的中间层,如图 1.1 所示。

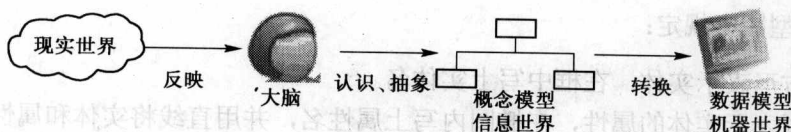


图 1.1 从现实世界到机器世界

概念模型主要是表示数据的逻辑特性,也就是说,概念模型只是在概念上表示数据库中将存储一些什么信息,而不管这些信息在数据库中是怎样实现存储的。概念模型描述事物、事物的特征和事物之间的联系,独立于计算机系统,即不依赖于硬件和软件。最常用的概念模型建立方法是实体-联系(E-R)模型,即 E-R 模型图。

1. 实体、属性与联系

1) 实体

在现实生活中,实体是指客观存在的、能够相互区分的事物。实体可以是人,也可以是物;可以指某个对象,也可以指某些概念。例如,学校、教师、学生、教室、桌椅等。一组特性相同的实体构成一个实体集。例如,全体学生就构成了一个实体集。

2) 属性

当使用计算机来描述实体时,采用的方法是针对这些实体的“特性”进行描述,所谓“特性”就是实体的“属性”。例如在大学环境里,学生是实体,不同的学生都具有学号、姓名、性别、出生日期、所属院系、专业、联系电话等属性。

3) 联系

正如现实世界中的事物不是孤立的一样,反映到信息世界中,实体内部和实体之间都存在着联系。实体之间的联系有以下 3 种类型。

- 一对一的联系(1:1)。对于实体集 A 中的每一个实体,在实体集 B 中如果只能找到一个实体和它对应,则这种联系称为一对一的联系。例如,对于“学校”实体集中的每一个学校,在“校长”实体集中只有一个校长和它对应,也就是说,一个学校只对应一个校长。
- 一对多的联系(1:M)。对于实体集 A 中的每一个实体,在实体集 B 中如果有若干个实体和它对应,则这种联系称为一对多的联系。例如,对于“教师”实体集中的每

一个教师，在“学生”实体集中可能有多个学生和它对应，也就是说，一个教师可能给多个学生上课。

- 多对多的联系 (M:N)。对于实体集 A 中的每一个实体，在实体集 B 中可能有若干个实体和它对应；反过来，对于实体集 B 中的每一个实体，在实体集 A 中可能有若干个实体和它对应，这种联系称为多对多的联系。例如，对于“课程”实体集中的每一门课程，在“学生”实体集中可能有多个学生和它对应，也就是说，一门课程可能有多个学生选择；反过来，对于“学生”实体集中的每一个学生，在“课程”实体集中可能有多门课程和它对应，也就是说，一个学生可以选择多门课程。

2. E-R 模型图

在 E-R 模型图中规定：

- (1) 用长方形表示实体，在框中写上实体名。
- (2) 用椭圆表示实体的属性，在椭圆内写上属性名，并用直线将实体和属性连接起来。
- (3) 用菱形表示实体间的联系，菱形框内写上联系名，并用直线将菱形和相关实体连接起来。例如，某学院教学管理的 E-R 模型图，如图 1.2 所示。

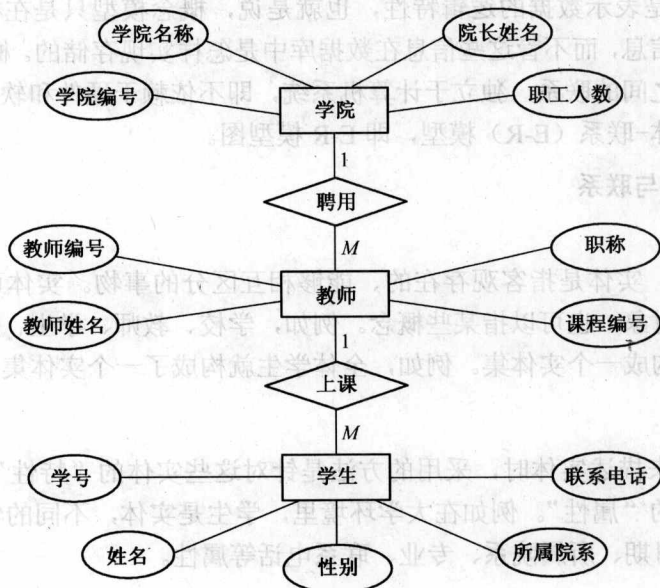


图 1.2 教学管理 E-R 模型图

1.2.2 实施模型

实施模型侧重于数据库中数据的表达方式和数据库结构的实现方法。实施模型主要有层次数据模型、网状数据模型和关系数据模型。基于这三种数据模型有相应的三种数据库系统，即层次数据库、网状数据库和关系数据库。

1. 层次数据模型

层次模型的设计思想是把系统划分成若干个小部分，然后再按照层次结构逐级组合成一个整体。层次结构从图的角度看好像一棵倒置的树，具有以下特点：

- 有且只有一个节点无双亲，这个节点称为根节点。
- 其他节点有且仅有一个双亲。

层次数据库的特点是将整个数据库作为一个整体存储，用指针实现数据之间的联系。例如，某大学的组织结构即为层次模型，如图 1.3 所示。

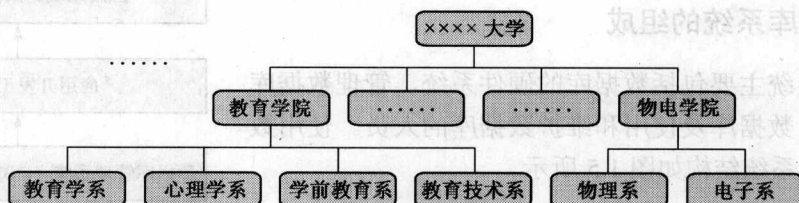


图 1.3 层次数据模型结构图

2. 网状数据模型

网状模型用图结构表示数据及数据之间的联系，网状模型具有以下特点：

- 允许节点无双亲，或有一个以上的双亲。
- 两个节点之间可以有多重关系。
- 一个节点有到自身的关系。

网状模型结构相当复杂，每个节点之间都可能发生联系。网状结构往往能更容易表示现实世界中事物间的复杂关系。例如，某学校不同事物之间关系的网状模型如图 1.4 所示。

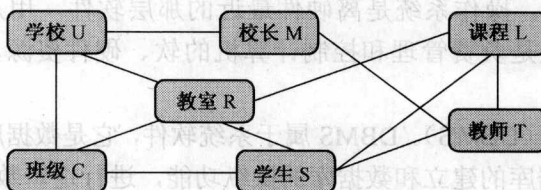


图 1.4 网状数据模型结构图

3. 关系数据模型

关系数据模型是三种数据模型中最重要的模型。在关系模型中，数据的逻辑结构是一张二维表。关系模型中的主要术语如下：

- **关系**：一个关系指一张二维表。
- **元组**：指二维表中除去表头行的每一行，一行称为一个元组。
- **属性**：指二维表中的每一列，包含属性名和属性值。
- **域**：指属性的取值范围。
- **主键**：指用来标识每个元组的单个属性或属性组合，其值具有唯一性。
- **逻辑结构**：是对关系的描述，用以下格式表示：

关系名(属性名 1, 属性名 2, 属性名 3, ..., 属性名 n)

例如，学生关系的逻辑结构为：学生(学号, 姓名, 性别, 所属院系, 专业, 电话)。