



普通高等教育“十三五”规划教材

高等学校计算机规划教材

# 数据库原理与应用

◆ 于 啸 陆丽娜 张 宇 主编  
◆ 白晨生 毕春光 张喜海 林 楠 副主编 ◆ 苏中滨 主审



中国工信出版集团



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
<http://www.phei.com.cn>

普通高等教育“十三五”规划教材

高等学校计算机规划

# 数据库原理与应用

于 啸 陆丽娜 张 宇 主编

白晨生 毕春光 张喜海 林 楠 副主编

丁宝峰 孙 建 姜 微 参编

苏中滨 主审

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书系统地介绍了数据库原理、SQL Server 2008 数据库管理系统及应用实例，概述了数据库前沿技术。全书共分 9 章，主要内容包括：绪论；关系数据库；SQL 语言；关系数据理论；数据库设计；数据库保护；数据库系统的访问；数据库技术的发展；SQL Server 2008 及应用实例，包括 SQL Server 2008 概述、SQL Server 2008 管理工具简介、Transact-SQL 语言基础、数据库管理、表的管理、视图的管理、存储过程、用户和安全性管理、数据转换服务、数据库应用开发实例等。

本书内容丰富、全面、系统，深度和广度兼顾，可作为高等院校和科研院所计算机专业和相关专业数据库课程的教材，也可作为有关人员学习和研究数据库原理与应用或开发数据库应用系统的技术参考书。

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

数据库原理与应用 / 于啸, 陆丽娜, 张宇主编. —北京: 电子工业出版社, 2017.8

ISBN 978-7-121-31465-0

I. ①数… II. ①于… ②陆… ③张… III. ①数据库系统—高等学校—教材 IV. ①TP311.13

中国版本图书馆 CIP 数据核字 (2017) 第 096004 号

策划编辑：戴晨辰

责任编辑：张 京

印 刷：北京季蜂印刷有限公司

装 订：北京季蜂印刷有限公司

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：17.5 字数：482 千字

版 次：2017 年 8 月第 1 版

印 次：2017 年 8 月第 1 次印刷

定 价：39.80 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, 88258888。

质量投诉请发邮件至 [zltts@phei.com.cn](mailto:zltts@phei.com.cn)，盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

本书咨询联系方式：[dcc@phei.com.cn](mailto:dcc@phei.com.cn)。

# 前 言

数据库技术产生于 20 世纪 60 年代, 经历了格式化数据库(以层次和网状数据库为代表)、经典数据库(以关系数据库和后关系数据库为代表)和新型数据库(以对象数据库和 XML 数据库等为代表)的三代发展演变。40 多年来, 数据库技术的重要性和意义已经被人们所认识与理解。首先, 数据库技术已经形成相对完整和成熟的科学理论体系, 成为现代计算机信息处理系统的重要基础与技术核心, 造就了 C. W. Bachman、E. F. Codd 和 J. Gray 三位图灵大奖得主; 其次, 数据库带动和形成了一个巨大的软件产业——数据库管理系统产品和相关技术工具与解决方案, 对经济发展起着极大的推动作用, 表现出非凡的生产力效应; 最后, 数据库研究和开发领域的各项成就推动了其他众多计算机理论与应用领域的进步, 对这些领域的发展起到了巨大的支撑作用, 成为各种计算机信息系统的核心内容与技术基础。

本书对数据库技术进行了全面的阐述和研究。在结合大量的实例和作者教学体会的基础上, 对数据库技术的各个领域进行了深入浅出的剖析, 对数据库技术的重点和难点进行了详细的描述, 力求做到思路清晰、概念准确、结构合理、内容生动活泼。本书重点介绍了数据库的基本概念及 ER 图、关系模型等数据库建模技术, 数据库应用部分主要介绍了 SQL Server 2008 的基本管理与操作。

每一章的开始都对该章将要涉及的内容及其作用进行了分析, 然后指出了学完本章读者应该掌握的重要内容。另外, 章末所附的练习题, 旨在加深读者对本章涉及概念的理解, 培养学生应用本章学到的知识来解决实际问题的能力。

本书内容全面、实例丰富, 并配备了课后习题参考答案, 方便教学。本书可作为高等院校计算机专业及信息管理等相关专业本科生数据库课程的教材, 也可作为相关人员学习数据库知识的参考书。

东北农业大学于啸、陆丽娜、张宇任本书主编; 沈阳工业大学白晨生, 吉林农业大学毕春光、林楠, 东北农业大学张喜海任本书副主编; 东北农业大学丁宝峰、孙建, 哈尔滨金融学院姜微参编; 全书由东北农业大学苏中滨教授主审。

本书的配套教学资源可在华信教育资源网([www.hxedu.com.cn](http://www.hxedu.com.cn))注册后免费下载。

由于作者水平有限, 加之创作时间仓促, 书中不足之处在所难免, 欢迎广大读者批评指正。

编 者

2017 年 5 月

# 目 录

第 1 章 绪论	1	2.5.2 关系数据库	23
1.1 数据库系统概述	1	2.6 关系运算	23
1.1.1 数据库的几个基本概念	1	2.6.1 关系代数	23
1.1.2 数据管理技术的发展历史	2	2.6.2 关系演算	28
1.2 数据模型	3	2.7 小结	28
1.2.1 信息的三个领域	3	第 3 章 SQL 语言	29
1.2.2 概念模型	4	3.1 SQL 语言概述	29
1.2.3 概念模型表示方法	5	3.1.1 SQL 语言的发展	29
1.3 数据模型	6	3.1.2 SQL 的基本概念	29
1.3.1 数据模型的组成要素	6	3.1.3 SQL 语言的主要特点	30
1.3.2 最常用的数据模型	7	3.2 数据定义	30
1.4 数据库的体系结构	12	3.2.1 基本表的定义、删除与修改	31
1.4.1 数据库的分级结构	12	3.2.2 索引的建立与删除	32
1.4.2 模式及映像	13	3.3 数据查询	33
1.5 数据库管理系统	14	3.3.1 单表查询	34
1.5.1 DBMS 的功能	14	3.3.2 连接查询	40
1.5.2 DBMS 的程序组成	15	3.3.3 嵌套查询	41
1.5.3 数据语言	16	3.3.4 集合查询	45
1.5.4 数据字典	17	3.4 数据操作	46
1.6 小结	17	3.4.1 数据插入	46
第 2 章 关系数据库	18	3.4.2 数据修改	47
2.1 从格式化模型到关系模型	18	3.4.3 数据删除	48
2.2 关系的数学定义	18	3.5 视图	49
2.2.1 域	18	3.5.1 定义视图	49
2.2.2 笛卡儿积	18	3.5.2 查询视图	50
2.2.3 关系的数学定义	19	3.5.3 更新视图	51
2.3 关系的性质	20	3.6 数据控制	52
2.4 码的概念	20	3.6.1 授权	52
2.4.1 码的定义	20	3.6.2 收回权限	53
2.4.2 候选码和主码	21	3.7 小结	54
2.4.3 外部关系码	21	第 4 章 关系数据理论	55
2.4.4 关系模型的完整性	21	4.1 关系规范化的作用	55
2.5 关系数据库模式	23	4.1.1 规范化理论概述	55
2.5.1 关系模式	23		

4.1.2	不合理的关系模式存在的 问题	55	5.2.2	需求收集	87
4.2	函数依赖	57	5.2.3	需求分析过程	88
4.2.1	关系模式的简化表示	57	5.2.4	需求分析的结果	90
4.2.2	函数依赖的基本概念	58	5.3	概念结构设计	90
4.2.3	码的函数依赖表示	59	5.3.1	概念结构设计概述	90
4.2.4	函数依赖和码的唯一性	60	5.3.2	概念结构设计的方法	91
4.3	规范化	60	5.3.3	局部 ER 图设计	91
4.3.1	规范化的含义	61	5.3.4	全局 ER 图设计	93
4.3.2	第一范式	61	5.4	逻辑结构设计	95
4.3.3	第二范式	62	5.4.1	概念模型转换为关系数据 模型	95
4.3.4	第三范式	63	5.4.2	关系模型的优化	96
4.3.5	BCNF 范式	64	5.4.3	设计用户子模式	97
4.4	多值依赖与 4NF	65	5.5	物理结构设计	97
4.4.1	问题的引入	65	5.5.1	物理结构设计概述	97
4.4.2	多值依赖基本概念	66	5.5.2	存取方法选择	98
4.4.3	第四范式	67	5.5.3	存储结构的确定	99
4.5	函数依赖的公理系统	68	5.6	数据库的实施和维护	99
4.5.1	函数依赖集的完备性	68	5.6.1	数据库的实施	99
4.5.2	函数依赖的推理规则	69	5.6.2	数据库的运行和维护	100
4.5.3	属性的闭包与 $F$ 逻辑蕴涵 的充要条件	70	5.7	小结	101
4.5.4	最小函数依赖集 $F_{\min}$	71	<b>第 6 章</b>	<b>数据库保护</b>	102
4.6	关系模式分解	73	6.1	数据库事务处理	102
4.6.1	无损分解	73	6.1.1	事务的定义	102
4.6.2	保持函数依赖	76	6.1.2	事务的 ACID 性质	103
4.7	连接依赖与 5NF	77	6.1.3	事务处理模型	103
4.7.1	连接依赖	77	6.1.4	事务的状态	104
4.7.2	第五范式	79	6.2	数据库故障及恢复	105
4.8	关系模式规范化步骤	79	6.2.1	数据库故障分类	105
4.9	小结	80	6.2.2	数据库恢复技术	106
6.2.3			6.2.3	数据库恢复策略	109
<b>第 5 章</b>	<b>数据库设计</b>	82	6.3	并发控制	110
5.1	数据库设计概述	82	6.3.1	并发操作带来的数据 不一致性	111
5.1.1	数据库设计的任务	82	6.3.2	封锁技术	112
5.1.2	数据库设计的特点	83	6.3.3	封锁带来的问题	114
5.1.3	数据库设计的方法	83	6.3.4	并发操作的调度	116
5.1.4	数据库设计的工具	84	6.4	数据库的完整性	117
5.1.5	数据库设计的步骤	85	6.4.1	数据的完整性	118
5.2	需求分析	86	6.4.2	SQL 中的完整性约束	118
5.2.1	需求分析的任务	86			

6.4.3	SQL 中的触发器	121	8.5	数据仓库与数据挖掘	162
6.5	数据库的安全性	123	8.5.1	数据仓库	162
6.5.1	数据库安全性概述	123	8.5.2	数据仓库设计实例	168
6.5.2	SQL 中的安全性机制	124	8.5.3	数据挖掘技术概述	170
6.5.3	常用的安全性措施	126	8.6	其他新型的数据库系统	172
6.6	小结	127	8.6.1	多媒体数据库系统	172
<b>第 7 章</b>	<b>数据库系统的访问</b>	129	8.6.2	主动数据库系统	173
7.1	数据库的访问接口	129	8.6.3	演绎数据库系统	173
7.1.1	固有调用	129	8.6.4	实时数据库系统	175
7.1.2	ODBC	129	8.7	小结	175
7.1.3	JDBC	133	<b>第 9 章</b>	<b>SQL Server 2008 及应用实例</b>	176
7.2	客户机/服务器模式的数据库系统	137	9.1	SQL Server 2008 概述	176
7.3	浏览器/服务器模式的数据库系统	139	9.1.1	概述	176
7.3.1	Web 数据库的体系结构	139	9.1.2	SQL Server 2008 的基本特点	176
7.3.2	技术实现	140	9.1.3	SQL Server 2008 的安装	176
7.4	Java 访问数据库的技术	142	9.2	SQL Server 2008 管理工具简介	182
7.4.1	JDBC 技术	142	9.2.1	创建服务器组	182
7.4.2	JSP 的数据库访问技术	142	9.2.2	注册服务器	182
7.5	数据库系统的多层体系结构	143	9.2.3	新建查询	183
7.6	小结	144	9.3	Transact-SQL 语言基础	184
<b>第 8 章</b>	<b>数据库技术的发展</b>	145	9.3.1	Transact-SQL 简介	184
8.1	数据库新技术的分类	145	9.3.2	数据类型	185
8.2	面向对象数据库系统	146	9.3.3	常量	187
8.2.1	面向对象程序设计方法	146	9.3.4	变量	187
8.2.2	面向对象数据模型	147	9.3.5	注释符和运算符	188
8.2.3	面向对象数据库语言	150	9.3.6	流程控制语句	188
8.2.4	面向对象数据库的模式演进	151	9.3.7	常用函数	190
8.2.5	对象-关系数据库	151	9.4	数据库管理	195
8.3	分布式数据库系统	152	9.4.1	创建数据库	196
8.3.1	分布式数据库系统的特点与目标	153	9.4.2	修改数据库设置	196
8.3.2	分布式数据库系统的体系结构	155	9.4.3	删除数据库	197
8.4	网络环境下的数据库体系	156	9.4.4	分离/附加数据库	198
8.4.1	客户机/服务器系统	156	9.4.5	备份/还原数据库	199
8.4.2	开放式数据库的互连技术 (ODBC)	159	9.5	表的管理	201
			9.5.1	创建数据表	201
			9.5.2	修改表	202
			9.6	视图的管理	205

9.6.1	创建视图	205	9.8.4	权限管理	220
9.6.2	管理视图	206	9.9	数据转换服务	223
9.7	存储过程	207	9.9.1	DTS 导出向导	224
9.7.1	创建存储过程	208	9.9.2	DTS 导入向导	225
9.7.2	执行存储过程	209	9.10	数据库应用开发实例	229
9.7.3	查看、修改和删除存储 过程	210	9.10.1	建立数据源	229
9.8	用户和安全性管理	212	9.10.2	创建新的工程	231
9.8.1	SQL Server 登录认证	212	9.11	小结	233
9.8.2	数据库用户	215	第 10 章	部分章节习题	234
9.8.3	角色管理	216	参考文献		270



# 第1章 绪论

20世纪60年代末,数据库技术初露头角,随即得到迅速发展,成为数据处理的公用支撑技术,是计算机科学的重要分支,也是信息系统的核心和基础。它的出现极大地促进了计算机应用向教育、科研、金融、医疗等行业的渗透。同时,数据库的建设规模、数据库信息量的大小和使用频度已成为衡量一个国家信息化程度的重要标志。

## 1.1 数据库系统概述

### 1.1.1 数据库的几个基本概念

#### 1. 数据(Data)

数据是数据库中存储的基本对象,是描述事物的符号记录。数据有多种表现形式,包括数字、文字、图像、声音、学生的档案记录、货物的运输情况等。为了方便计算机存储和处理日常生活中的事物,通常抽象出事物的特征并将其组成一个记录来描述。例如,在教师档案中,用姓名、性别、出生年份、所在系别、工作时间描述一个教师的基本情况,用记录形式表示为:(宋国,男,1998,信息管理系,2016)。这里的教师记录就是数据。结合其含义就能得到该教师的个人情况:宋国是一个教师,性别男,1998年出生,2016年进入信息管理系。如果不了解其语义,则无法理解该记录含义。可见,数据的形式还不能完全表达其内容,需要对数据进行解释。所以数据和关于数据的解释是不可分的,数据的解释是指对数据含义的说明,数据的含义称为数据的语义,数据与其语义是不可分的。

#### 2. 数据库(DataBase, DB)

数据库是长期存储在计算机存储设备上的、有组织的、可共享的数据集合。数据库中的数据按一定的数据模型组织、描述和储存,具有较小的冗余度、较高的数据独立性和易扩充性,并可共享。

#### 3. 数据库管理系统(DataBase Management System, DBMS)

数据库管理系统是位于用户与操作系统之间的一层数据管理软件,研究如何科学地组织和存储数据、如何高效地获取和维护数据。它是数据库系统的一个重要组成部分,主要功能如下。

##### 1) 数据定义功能

DBMS提供数据定义语言(Data Definition Language, DDL),用户通过它可以方便地对数据库中的数据对象进行定义。

##### 2) 数据操纵功能

DBMS提供数据操纵定义语言(Data Manipulation Language, DML),用户通过它实现对数据的基本操作,如查询、插入、删除和修改等。

### 3) 数据库的运行管理

数据库在建立、运行和维护时由数据库管理系统统一管理、统一控制,以保证数据的安全性、完整性、多用户对数据的并发使用及发生故障后的系统恢复。

### 4) 数据库的建立和维护功能

数据库的建立和维护功能包括数据库初始数据的输入、转换功能,数据库的转储、恢复功能,数据库的重组、重构功能和性能监视、分析功能等。

## 4. 数据库系统(DataBase System, DBS)

数据库系统是指在计算机系统中引入数据库后的系统,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(DataBase Administrator, DBA)和用户构成。数据库系统如图 1-1 所示。数据库在计算机系统的位置如图 1-2 所示。

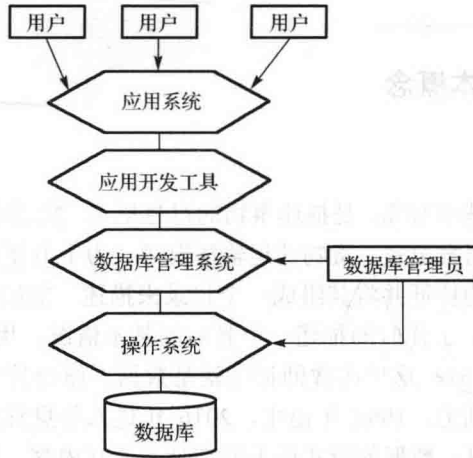


图 1-1 数据库系统

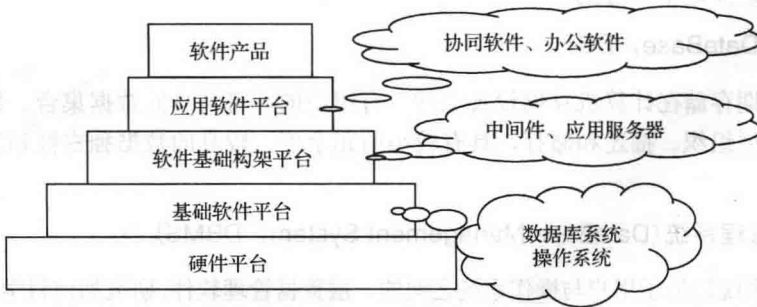


图 1-2 数据库在计算机系统的位置

### 1.1.2 数据管理技术的发展历史

数据库技术是应数据管理任务的需要而产生的。数据处理是指对各种数据进行收集、存储、加工和传播的一系列活动的总和。数据管理则是指对数据进行分类、组织、编码、存储、检索和维护,它是数据处理的中心问题。数据管理技术经历了人工管理(20 世纪 40 年代中—50 年代中)、文件系统(20 世纪 50 年代末—60 年代中)、数据库系统(20 世纪 60 年代末—现在)三个阶段。这三个阶段的特点及其比较如表 1-1 所示。

表 1-1 数据管理三个阶段比较表

		人工管理阶段	文件系统阶段	数据库系统阶段
背景	应用背景	科学计算	科学计算、管理	大规模管理
	硬件背景	无直接存取存储设备	磁盘的、磁鼓	大容量磁盘
	软件背景	没有操作系统	有文件系统	有数据库管理系统
	处理方式	批处理	联机实时处理、批处理	联机实时处理、分布处理、批处理
特点	数据的管理者	用户(程序员)	文件系统	数据库管理系统
	数据面向的对象	某一应用程序	某一应用	现实世界
	数据的共享程度	无共享, 冗余度极大	共享性差, 冗余度大	共享性高, 冗余度小
	数据的独立性	不独立, 完全来自于程序	独立性差	具有高度的物理独立性和一定的逻辑独立性
	数据的结构化	无结构	记录内有结构、整体无结构	整体结构化, 用数据模型描述
	数据控制能力	应用程序自己控制	应用程序自己控制	由数据库管理系统提供数据安全性、完整性、并发控制和恢复能力

## 1.2 数据模型

数据库技术是计算机领域中发展最快的技术之一。数据库技术的发展是沿着数据模型的主线推进的。模型,特别是具体模型对人们来说并不陌生。一张地图、一组建筑设计沙盘、一架精致的航模飞机都是具体的模型,都是一眼望去就会使人联想到真实生活的事物。模型是对现实世界中某个对象特征的模拟和抽象。

数据模型也是一种模型,它是对现实世界数据特征的抽象。通俗地讲数据模型就是现实世界的模拟。

由于计算机不可能直接处理现实世界中的具体事物,所以人们必须事先把具体事物转换成计算机能处理的数据,也就是首先要数字化,把现实世界中的具体的人、物、活动、概念用数据模型这个工具来抽象、表示和处理。

数据模型是数据库系统的核心和基础。现有的数据库系统均是基于某种模型的。根据模型应用的不同目的,可分为两类:第一类是概念模型,也称信息模型,按用户的观点来对数据和信息进行建模,主要用于数据库设计;另一类模型是数据模型,包括网状模型、层次模型、关系模型等,它按计算机系统的观点对数据建模,主要用于数据库系统的实现。

### 1.2.1 信息的三个领域

在现实世界中,信息处于三个领域:现实世界、信息世界和机器世界。①现实世界是存在于人们头脑之外的客观世界,事物及其相互联系就处于这个世界中。②信息世界是现实世界在人们头脑中的反映。客观事物在信息世界中被称为实体,反映事物联系的是实体模型。③机器世界是信息世界中信息的数据化。现实世界中的事物及联系在这里用数据模型描述。信息所处三个领域的联系如图 1-3 所示。

可见,客观事物是信息之源,是设计数据库的出发点,也是使用数据库的最终归宿。实体模型与数据模型是对客观事物及其联系的两级抽象描述,数据库的核心问题是数据模型,数据模型由实体模型导出。

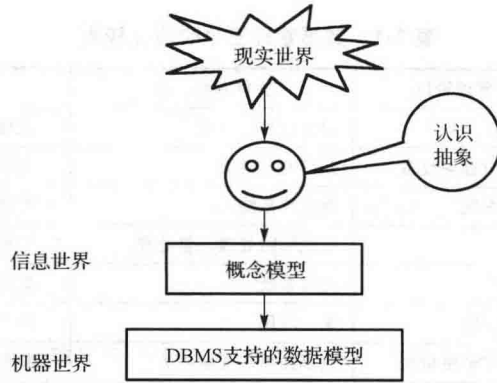


图 1-3 现实世界、信息世界和机器世界之间的联系

## 1.2.2 概念模型

由图 1-3 可以看出,概念模型实际上是现实世界到机器世界的一个中间层次。概念模型也称信息模型,它是按用户的观点来对数据和信息建模,用于数据库设计。它是现实世界到机器世界的一个中间层次,是数据库设计的有力工具,是数据库设计人员和用户之间进行交流的语言。

在信息世界主要涉及以下一些概念。

### 1. 实体(Entity)

实体是客观世界中存在的且可相互区分的事物,实体可以是人、事、物,也可以是抽象概念或联系。例如,一个职工、一个老师、一个学生、一门课、学生的一次选课、老师与院系的工作关系等都是实体,在 E-R 图中用矩形框表示。

### 2. 属性(Attribute)

实体所具有的某一特性称为属性。一个实体可以由若干个属性来刻画。例如,学生实体可以由学号、姓名、性别、年龄、系别等属性组成。

### 3. 码(Key)

唯一标识实体的属性集称为码。例如,学号是学生实体码。

### 4. 域(Domain)

属性的取值范围称为该属性的域。例如,学号的域为 8 位整数,姓名为字符串集合。

### 5. 实体型(Entity Type)

用实体名及其属性名集合来抽象和刻画同类实体称为实体型。例如,学生(学号、姓名、性别、年龄、系别)就是一个实体型。

### 6. 实体集(Entity Set)

同一类型实体的集合称为实体集。例如,全体学生就是一个实体集。

### 7. 联系(Relationship)

现实世界中事物内部及事物之间的联系在信息世界中反映为实体内部的联系和实体之间的

联系。实体内部的联系通常是指组成实体的各属性之间的联系。实体之间的联系通常是指不同实体集之间的联系。

两个实体型之间的联系可以分为三类。

### 1) 一对一联系 (1:1)

如果对于实体集  $A$  中的每一个实体，实体集  $B$  中至多有一个(也可以没有)实体与之联系，反之亦然，则称实体集  $A$  与实体集  $B$  具有一对一联系，记为 1:1。例如乘客与车票、病人与床位之间的联系。

### 2) 一对多联系 (1:n)

如果对于实体集  $A$  中的每一个实体，实体集  $B$  中有  $n$  个实体 ( $n \geq 0$ ) 与之联系，反之，对于实体集  $B$  中的每一个实体，实体集  $A$  中至多只有一个实体与之联系，则称实体集  $A$  与实体集  $B$  具有一对多联系，记为 1:n。例如，班级与学生之间具有一对多联系。

### 3) 多对多联系 (m:n)

如果对于实体集  $A$  中的每一个实体，实体集  $B$  中有  $n$  个实体 ( $n \geq 0$ ) 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  $m$  个实体 ( $m \geq 0$ ) 与之联系，则称实体集  $A$  与实体集  $B$  具有多对多联系，记为  $m:n$ 。例如，学生与课程之间具有多对多联系。

可以用图形来表示两个实体型之间的这三类关系，如图 1-4 所示。

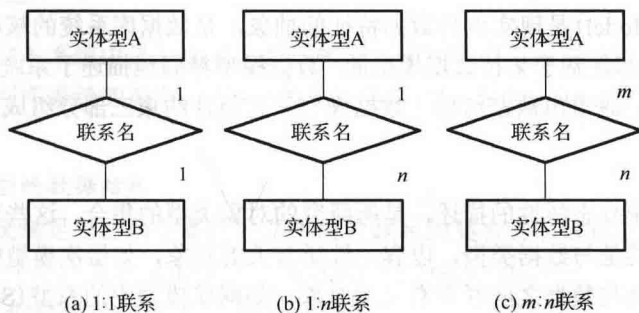


图 1-4 两个实体型之间的三类关系

## 1.2.3 概念模型的表示方法

概念模型是对信息世界建模，所以概念模型能够方便、准确地表示信息世界中的常用概念。概念模型的表示方法很多，其中最为著名、最为常用的是 P.P.S.Chen 于 1976 年提出的实体-联系方法(Entity-Relationship Approach)。该方法用 E-R 图来描述现实世界的概念模型，E-R 方法也称为 E-R 模型。

E-R 图提供了表示实体型、属性和联系的方法。

① 实体型：用矩形表示，矩形框内写明实体名。

② 属性：用椭圆形表示，并用无向边将其与相应的实体连接起来。

③ 联系：联系本身用菱形表示，菱形框内写明联系名，并用无向边分别与有关实体连接起来，同时，在无向边旁标上联系的类型(1:1、1:n 或  $m:n$ )。

需要注意的是，如果一个联系具有属性，则这些属性也要用无向边与该联系连接起来。

现以某校教学管理为例建立实体模型，E-R 图如图 1-5 所示。

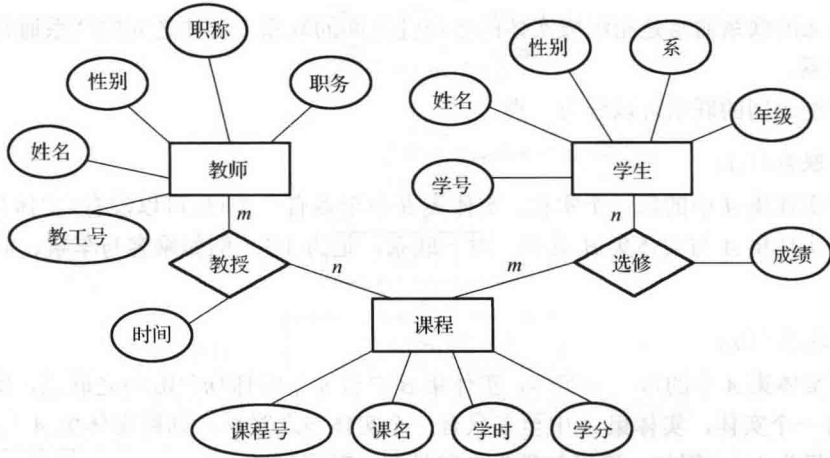


图 1-5 某学校教学管理 E-R 图

## 1.3 数据模型

### 1.3.1 数据模型的组成要素

数据模型(Data Model)是现实世界数据特征的抽象,是数据库系统的核心和基础。各种机器上实现的 DBMS 软件都是基于某种数据模型的。数据模型精确地描述了系统的静态特性、动态特性和完整性约束条件,通常由数据结构、数据操作和完整性约束三部分组成。

#### 1. 数据结构

数据结构是对系统静态特性的描述,是所研究的对象类型的集合。这些对象是数据库的组成部分,包括两类:一类是与数据类型、内容、性质有关的对象,如层次模型中的数据项、关系模型中的关系;另一类是与数据之间联系有关的对象,如网状模型中的系型(Set Type)。

数据结构是刻画一个数据模型性质最重要的方面。因此在数据库系统中,通常按照数据结构的类型来命名数据模型。例如,层次结构的数据模型命名为层次模型,关系结构的数据模型命名为关系模型。

#### 2. 数据操作

数据操作是对系统动态特性的描述,是对数据库中各种对象(型)的实例(值)允许执行的操作的集合,包括操作及有关的操作规则。数据操作主要有检索和更新(包括插入、删除、修改)两大类操作。数据模型必须定义这些操作的确切含义、操作符号、操作规则及实现操作的语言。

#### 3. 数据的约束条件

数据的约束条件是一组完整性规则的集合。完整性规则是给定的数据模型中数据及其联系所具有的制约和储存规则,用以限定符合数据模型的数据库状态及状态的变化,以保证数据正确、有效、相容。

数据模型应该反映和规定本数据模型必须遵守的、基本的、通用的完整性约束条件。例如,在关系模型中,任何关系必须满足实体完整性和参照完整性两个条件。此外,数据模型还应该提供定义完整性约束条件的机制,以反映具体应用所涉及的数据必须遵守的特定的语义约束条件。

### 1.3.2 最常用的数据模型

目前，数据库领域中最常用的数据模型有五种，它们是：

- 层次模型(Hierarchical Model)；
- 网状模型(Network Model)；
- 关系模型(Relational Model)；
- 面向对象模型(Object Oriented Model)；
- 对象关系模型(Object Relational Model)。

其中层次模型和网状模型统称为非关系模型。

非关系模型的数据库系统盛行于 20 世纪 70 年代至 80 年代初。在非关系模型中，实体用记录表示，实体的属性对应记录的数据项(或字段)。实体之间的联系在非关系模型中转换成记录之间的两两联系。非关系模型中数据结构的单位是基本层次联系(见图 1-6)。基本层次联系：两个记录及它们之间的一对多(包括一对一)的联系。

20 世纪 80 年代以来，面向对象的方法和技术对计算机各个领域都产生了深远的影响，促进了数据库中面向对象数据模型的研究和发展。

下面简要介绍层次模型、网状模型和关系模型。

#### 1. 层次模型

层次模型是数据库系统中最早出现的数据模型，层次模型用树形结构来表示各类实体及实体之间的联系。层次数据库系统的典型代表是 IBM 公司的 IMS (Information Management System) 数据库管理系统。

##### 1) 层次数据模型的数据结构

在数据库中定义满足下面两个条件的基本层次联系的集合为层次模型。

- ① 有且只有一个结点没有双亲结点，这个结点称为根结点；
- ② 根以外的其他结点有且只有一个双亲结点。

在层次模型中，每个节点表示一个记录类型，记录之间的联系用节点之间的连线(有向边)表示，这种联系是父子之间的一对多的联系。这就使得层次数据库系统只能处理一对多的实体联系。

每个记录型可以包含若干个字段，这里记录类型描述的是实体，字段描述的是实体的属性。各个记录类型可以定义一个排序字段，也称为码字段，如果定义该排序字段的值是唯一的，则它能唯一地表示一个记录值。在层次模型中，同一双亲的子女结点称为兄弟结点(Twin 或 Sibling)，没有子女结点的结点称为叶结点。图 1-7 是一个教员学生层次数据库模型。该层次数据库有四个记录型。记录型“系”是根结点，由“系编号”、“系名”、“办公地点”三个字段组成。它有两个子女“教研室”和“学生”。记录型“教研室”是“系”的子女结点，同时又是“教员”的双亲结点，它由“教研室编号”、“教研室名”两个字段组成。记录类型“学生”由“学号”、“姓名”、“成绩”三个字段组成。记录“教员”由“职工号”、“姓名”、“研究方向”三个字段组成。“学生”与“教员”是叶结点，它们没有子女结点。由“系”到“教研室”、由“教研室”到“教员”、由“系”到“学生”均是一对多关系。

##### 2) 多对多联系在层次模型中的表示

因为层次数据模型只能表示一对多(包括一对一)的联系，现实世界中大部分是多对多联系，在层次模型中采用分解法表示多对多联系，分解法有两种：冗余结点法和虚拟结点法。

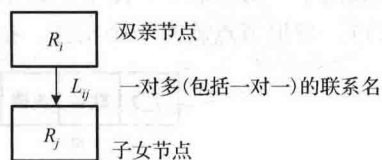


图 1-6 基本层次联系

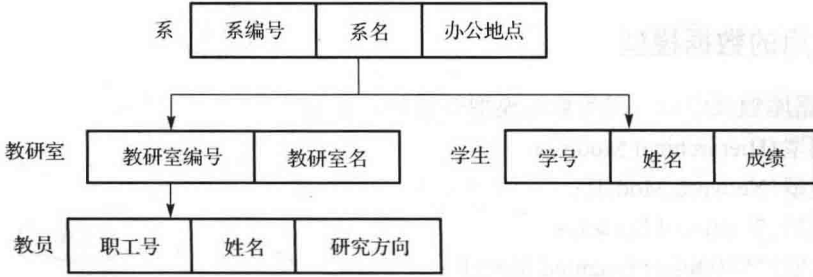


图 1-7 教员学生层次数据库模型

图 1-8(a) 是一个简单的多对多联系：一个学生可以选修多门课程，一门课程可由多个学生选修。“学生”字段由“学号”、“姓名”、“成绩”三个字段组成，“课程”由“课程号”和“课程名”两个字段组成。图 1-8(b) 采用冗余节点法，即通过增设两个冗余节点将图 1-8(a) 的多对多联系转换成两个一对多联系，图 1-8(c) 采用虚拟节点分解方法，即将图 1-8(b) 中的冗余节点转换为虚拟节点。虚拟节点就是一个指针，指向所替代的节点。

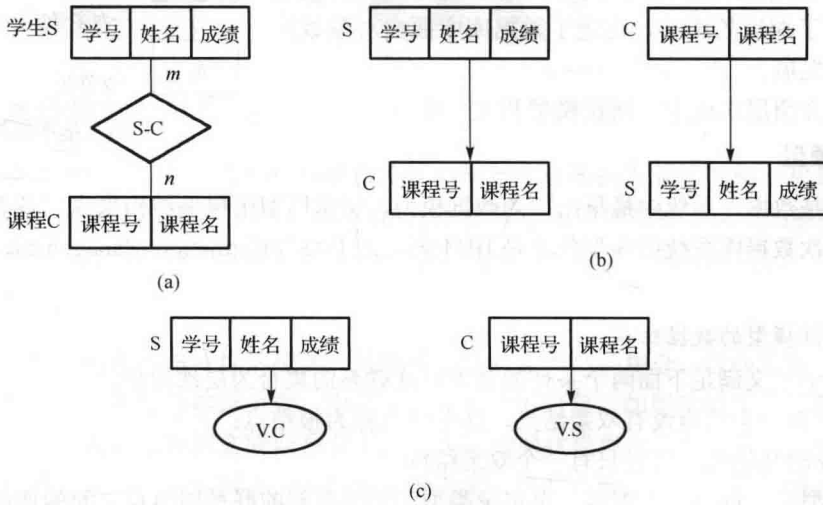


图 1-8 用层次模型表示多对多联系

### 3) 层次模型的数据操纵与完整性约束

层次模型的数据操纵主要有查询、插入、删除和修改。层次模型的完整性约束条件如下：

- (1) 无相应的双亲结点值就不能插入子女结点值；
- (2) 如果删除双亲结点值，则相应的子女结点值也被同时删除；
- (3) 更新操作时，应更新所有的相应记录，以保证数据的一致性。

### 4) 层次数据模型的存储结构

层次数据库中不仅要存储数据本身，还要存储数据之间的层次联系。常用的实现方法有以下两种。

#### (1) 邻接法

按照层次树前序穿越的顺序把所有记录值依次邻接存放，即通过物理空间的位置相邻来实现（或隐含）层次顺序。例如，对于图 1-9(a) 的数据库，按邻接法存放图 1-9(b) 中以根记录 A1 为首的层次记录实例集，应按图 1-10 所示的方法存放。



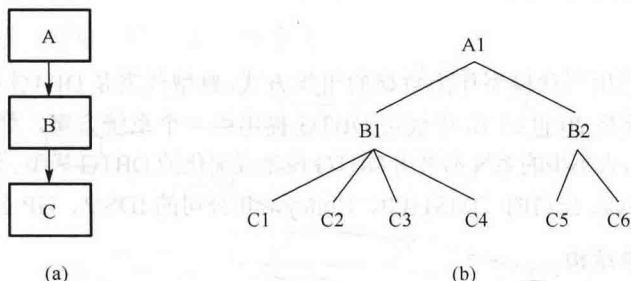


图 1-9 层次数据库及其实例

A1	B1	C1	C2	C3	C4	B2	C5	C6
----	----	----	----	----	----	----	----	----

图 1-10 邻接法

(2) 链接法

链接法用指引元来反映数据之间的层次联系。如图 1-11 所示，图 1-11 (a) 中的每个记录设两类指引元，分别指向最左边的子女和最近的兄弟，这种连接方法称为子女-兄弟链接法；图 1-11 (b) 按树的前序穿越顺序链接各记录值，这种链接方法称为层次序列链接法。

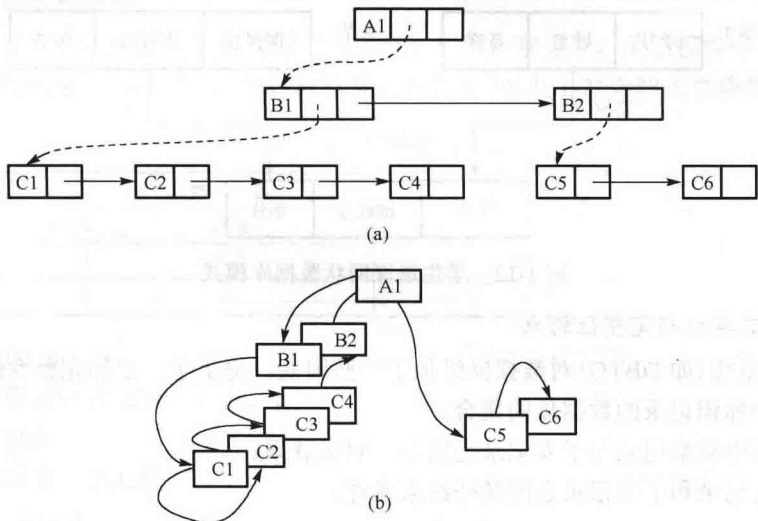


图 1-11 链接法

5) 层次模型的优缺点

层次模型的优点主要有：

- (1) 层次模型的数据结构简单清晰；
- (2) 查询效率高，性能优于关系模型，不低于网状模型；
- (3) 层次数据模型提供了良好的完整性支持。

层次模型的缺点主要有：

- (1) 多对多联系表示不自然；
- (2) 对插入和删除操作的限制多，应用程序的编写比较复杂；
- (3) 查询子女结点时必须通过双亲结点；
- (4) 由于结构严密，层次命令趋于程序化。