

21世纪高等教育计算机规划教材

COMPUTER

数据库技术与应用 教程 (Access)

Database Technology (Access)

张基温 文明瑶 丁群 朱莎 方晓 编著

采用案例组织形式、任务驱动方式

将数据库知识渗透在应用中

注重实践，突出能力培养



人民邮电出版社
POSTS & TELECOM PRESS

21世纪高等教育计算机规划教材

COMPUTER

数据库技术与应用 教程 (Access)

Database Technology (Access)

■ 张基温 文明瑶 丁群 朱莎 方晓 编著



人民邮电出版社

北 京

图书在版编目 (C I P) 数据

数据库技术与应用教程 : Access / 张基温等编著

— 北京 : 人民邮电出版社, 2013. 2

21世纪高等教育计算机规划教材

ISBN 978-7-115-30218-2

I. ①数… II. ①张… III. ①关系数据库系统—高等学校—教材 IV. ①TP311.138

中国版本图书馆CIP数据核字(2013)第003288号

内 容 提 要

本书介绍数据库技术及关系型数据库的基础知识,并以 Microsoft Access 2003 为背景,介绍 Access 数据库的创建和使用方法。全书以一个“考生数据库”为例贯穿始终,通过连续性的、功能丰富的实例介绍 Access 数据库各个对象的操作方法,包括数据表、查询、窗体、报表、数据访问页、宏和模块。最后以“超市管理系统”为例,综合、细致地介绍数据库系统的设计和开发过程。

本书注重实践,突出能力培养,采用案例组织形式、任务驱动方式,把数据库知识渗透在应用中;内容符合“全国计算机等级考试(二级 Access)”的考试大纲。适合作为普通高等院校及高职高专院校的计算机基础课程教材,尤其适合应用型高等院校学生使用,也可以作为相关培训班及计算机等级考试辅导班的教材或参考书。

21 世纪高等教育计算机规划教材

数据库技术与应用教程 (Access)

◆ 编 著 张基温 文明瑶 丁 群 朱 莎 方 晓
责任编辑 李海涛

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京昌平百善印刷厂印刷

◆ 开本: 787×1092 1/16
印张: 14.75 2013 年 2 月第 1 版
字数: 383 千字 2013 年 2 月北京第 1 次印刷

ISBN 978-7-115-30218-2

定价: 29.80 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223
反盗版热线: (010)67171154

目前,在高等院校非计算机专业中大面积开设了 Access 数据库技术与应用课程。Access 是 Microsoft 公司发布的 Office 系列软件中的一个重要成员。其操作简单、易学易用、界面友好、功能强大,不仅成为众多数据库初学者的首选,也在各类数据处理系统中得到了广泛应用。

第1章主要介绍数据库技术及关系型数据库的基础知识、数据库设计的方法与步骤,以及 Access 2003 数据库的特点及功能等。

第3章介绍查询的基本知识,并通过丰富的实例介绍五大查询的操作方法。

第5章主要介绍报表的基本知识、各种创建报表和编辑报表的方法,并通过实例介绍了报表的排序与分组、报表计算,最后简要说明打印报表的基本

第7章主要介绍宏的基本概念、宏的创建以及运行方法等。

第9章简要介绍 VBA 的数据库编程。

随着我国高等教育发展与改革的逐步深化,各类型高等院校的人才培养目标也逐步明确。对于应用型高等院校而言,重点是培养理论够用、实践能力强、能够应付各行各业需求的应用型人才。本书针对应用型高等院校的特点,采用案例驱动的编写方式,形成以案例为核心,以任务为线索,在介绍操作中渗透数据库知识体系结构。全书以一个“考生数据库”为例贯穿始终,通过连贯、丰富的实例介绍 Access 数据库各种对象的操作方法,包括数据表、查询、窗体、报表、数据访问页、宏和模块。

目 录

第 1 章 绪论	1	第 2 章 数据表	30
1.1 数据与数据模型	1	2.1 创建数据表	30
1.1.1 信息、数据与数据模型的概念	1	2.1.1 使用向导创建表	30
1.1.2 数据的概念模型	2	2.1.2 字段与数据类型	32
1.1.3 数据的逻辑模型	4	2.1.3 使用设计器创建表	33
1.2 数据库与数据库系统	5	2.1.4 通过输入数据创建表	35
1.2.1 数据管理技术的产生与发展	5	2.1.5 使用导入创建表	36
1.2.2 数据库及其特点	7	2.1.6 向表中输入数据	37
1.2.3 数据库管理系统与数据库 应用系统	8	2.1.7 字段属性	38
1.3 关系数据库	9	2.2 修改数据表结构	43
1.3.1 关系数据库概述	9	2.2.1 修改字段	43
1.3.2 关系的完整性	10	2.2.2 插入新字段	43
1.3.3 关系运算	10	2.2.3 移动字段	44
1.3.4 从 E-R 模型到关系表的转换	13	2.2.4 复制字段	44
1.4 Access 2003 简介	15	2.2.5 删除字段	44
1.4.1 Access 的发展	16	2.3 编辑数据表记录	46
1.4.2 Access 2003 的特点和功能	16	2.4 调整表外观	46
1.4.3 Access 2003 的启动与退出	16	2.4.1 调整字段显示高度和宽度	46
1.4.4 Access 2003 工作环境	18	2.4.2 设置数据表格式	46
1.5 Access 数据库的组成	19	2.4.3 改变字体显示	47
1.5.1 表对象	19	2.4.4 显示隐藏冻结列	47
1.5.2 查询对象	19	2.5 操作表	48
1.5.3 窗体对象	19	2.5.1 数据的查找与替换	48
1.5.4 报表对象	19	2.5.2 数据的排序与筛选	48
1.5.5 数据访问页对象	20	2.6 建立表之间的关系	52
1.5.6 宏	20	2.6.1 创建表关系	52
1.5.7 模块	20	2.6.2 主表与子表	54
1.6 创建 Access 数据库	20	2.7 本章小结	54
1.6.1 Access 数据库的创建	20	2.8 练习	54
1.6.2 Access 数据库的基本操作	26	第 3 章 查询	57
1.7 本章小结	26	3.1 查询的功能与类型	57
1.8 练习	26	3.1.1 查询的定义	57

3.1.2 查询的视图.....	57	4.6 练习.....	114
3.1.3 查询的类型及其功能.....	58	第 5 章 报表	115
3.2 选择查询.....	58	5.1 认识报表.....	115
3.2.1 创建选择查询的几种方法.....	58	5.1.1 报表的作用.....	115
3.2.2 查询的计算.....	65	5.1.2 报表的类型.....	115
3.3 参数查询.....	65	5.1.3 报表的视图.....	116
3.3.1 单参数查询.....	66	5.2 使用向导创建报表.....	117
3.3.2 多参数查询.....	66	5.2.1 自动创建报表.....	117
3.4 交叉表查询.....	67	5.2.2 使用向导创建报表.....	118
3.4.1 交叉表查询的作用.....	67	5.3 利用设计视图创建报表.....	123
3.4.2 创建交叉表查询的方法.....	67	5.3.1 报表设计视图的组成.....	123
3.5 操作查询.....	70	5.3.2 在设计视图中创建报表.....	123
3.5.1 生成表查询.....	70	5.3.3 创建主/子报表.....	124
3.5.2 更新查询.....	72	5.4 编辑报表.....	127
3.5.3 删除查询.....	73	5.4.1 为报表添加分页符及 直线、矩形.....	127
3.5.4 追加查询.....	75	5.4.2 自动套用格式.....	128
3.6 SQL 查询.....	76	5.4.3 为报表添加背景图案.....	129
3.6.1 SQL 的语法规则.....	77	5.4.4 为报表添加日期/时间和页码.....	130
3.6.2 创建 SQL 查询.....	78	5.5 报表的排序和分组.....	131
3.7 本章小结.....	84	5.5.1 记录排序.....	131
3.8 练习.....	84	5.5.2 记录分组.....	132
第 4 章 窗体	87	5.6 报表计算.....	134
4.1 认识窗体.....	87	5.6.1 计算控件的使用.....	134
4.1.1 窗体的作用.....	87	5.6.2 报表的统计计算.....	135
4.1.2 窗体的类型.....	87	5.7 打印报表.....	136
4.1.3 窗体的视图.....	89	5.7.1 页面设置.....	136
4.2 使用“向导”创建窗体.....	90	5.7.2 打印报表.....	138
4.2.1 自动创建窗体.....	90	5.8 本章小结.....	139
4.2.2 使用向导创建窗体.....	92	5.9 练习.....	139
4.3 利用“设计视图”创建窗体.....	96	第 6 章 数据访问页	141
4.3.1 窗体设计视图的组成.....	96	6.1 认识数据访问页.....	141
4.3.2 工具箱的使用.....	97	6.1.1 静态 HTML 页和动态 HTML 页.....	141
4.4 美化窗体.....	110	6.1.2 数据访问页的视图.....	141
4.4.1 窗体及控件的属性设置.....	110	6.2 创建数据访问页.....	142
4.4.2 窗体中控件的对齐及 尺寸设置.....	110	6.2.1 导出静态网页.....	142
4.4.3 添加日期/时间及页码.....	111	6.2.2 自动创建数据访问页.....	144
4.4.4 自动套用格式.....	112		
4.5 本章小结.....	113		

6.2.3 使用向导创建数据访问页	144	8.3.4 变量与常量	165
6.2.4 使用“设计视图”创建 数据访问页	146	8.3.5 常用标准函数	167
6.3 编辑数据访问页	146	8.4 VBA 流程控制语句	171
6.3.1 为数据访问页添加标题	146	8.4.1 赋值语句	172
6.3.2 为数据访问页添加常见控件	147	8.4.2 条件语句	172
6.3.3 为数据访问页设置 主题和背景	150	8.4.3 循环语句	175
6.4 本章小结	151	8.4.4 其他语句——标号和 Goto 语句	178
6.5 练习	151	8.5 过程调用和参数传递	178
第 7 章 宏	153	8.5.1 过程调用	178
7.1 认识宏	153	8.5.2 参数传递	179
7.1.1 宏的概念	153	8.6 VBA 程序运行错误处理	180
7.1.2 宏设计窗口	153	8.7 VBA 程序的调试	181
7.1.3 常用的宏操作	154	8.8 本章小结	182
7.2 宏的创建	155	8.9 练习	182
7.2.1 操作序列宏	155	第 9 章 VBA 数据库编程	189
7.2.2 条件操作宏	156	9.1 VBA 常见操作	189
7.2.3 宏组	157	9.1.1 打开和关闭操作	189
7.3 运行宏	158	9.1.2 输入框	189
7.3.1 直接运行宏	158	9.1.3 消息框	190
7.3.2 自动运行宏	158	9.1.4 VBA 编程验证数据	190
7.3.3 事件触发	158	9.1.5 计时事件 (Timer)	191
7.4 本章小结	159	9.1.6 鼠标和键盘事件处理	191
7.5 练习	159	9.2 VBA 的数据库编程	191
第 8 章 模块与 VBA 编程基础	160	9.2.1 数据库引擎及其接口	192
8.1 模块的基本概念	160	9.2.2 VBA 访问的数据库类型	192
8.1.1 类模块	160	9.2.3 数据访问对象 (DAO)	192
8.1.2 标准模块	160	9.2.4 ActiveX 数据对象 (ADO)	192
8.1.3 将宏转换为模块	160	9.3 本章小结	194
8.2 创建模块	160	9.4 练习	194
8.2.1 在模块中加入过程	160	第 10 章 超市进销存管理系统 开发综合示例	197
8.2.2 在模块中执行宏	161	10.1 系统分析与设计	197
8.3 VBA 程序设计基础	161	10.1.1 问题描述	197
8.3.1 面向对象程序设计的 基本概念	161	10.1.2 系统需求分析	197
8.3.2 Visual Basic 编辑环境	162	10.1.3 系统功能描述	198
8.3.3 数据类型和数据库对象	163	10.2 数据库概念结构设计	199
		10.3 数据库的创建	199

10.3.1 创建数据库.....	199	10.6 报表的设计.....	215
10.3.2 创建数据表.....	200	10.6.1 基础数据报表.....	215
10.3.3 建立表间的关系.....	201	10.6.2 数据统计分析报表.....	216
10.3.4 输入或导入数据.....	202	10.7 菜单与宏的设计.....	219
10.4 查询的设计与创建.....	202	10.8 系统设置与编译运行.....	220
10.4.1 基本信息查询.....	203	10.9 本章小结.....	220
10.4.2 计算统计与交叉表查询.....	205	10.10 练习.....	221
10.5 窗体的设计.....	208	附录 2012 年全国 Access	
10.5.1 信息浏览窗体.....	208	二级考试大纲.....	222
10.5.2 数据操作窗体.....	209	参考文献.....	226
10.5.3 查询统计窗体.....	210		
10.5.4 系统交互窗体.....	213		

第 1 章

绪论

今天,凡是要处理大量数据的地方,譬如学校对学生档案及成绩的管理,银行对各种业务的管理,企业对财务等数据的管理,酒店对客房的管理等,无一不用到数据库技术,数据库的规模、它所存储的信息量大小和使用频度已成为衡量时代信息化程度的重要标志。

本章主要讲述数据、信息、数据处理、数据库、数据库管理系统等数据库系统的相关概念,以及数据模型、关系数据库、数据库设计、Access 数据库等数据库技术基础知识。

1.1 数据与数据模型

1.1.1 信息、数据与数据模型的概念

1. 信息与数据

在信息时代,引用和出现频率最高、联系极为密切的两个名词是:数据(data)与信息。关于它们,目前还没有公认严格定义。但是应该说,它们是紧密联系而又有区别的。要说它们没有联系,那么为什么有时称信息处理,有时又称数据处理呢?要说它们没有区别,那为什么“数据库”(data base)技术不叫做“信息库”(information base)技术呢?计算机处理的到底是数据(data)还是信息(information)呢?数据与信息到底有什么区别呢?对此数据处理的学术界感到有必要对两者进行界定。我国著名学者萨师煊教授(1922—2010年,见图1.1)认为:数据是记录下来可以被鉴别的符号,信息是对数据的解释。例如,数据可以用图形、图像、声音、文字等不同的形式表现,用光、声、电、磁、纸张承载,只要可以鉴别,就称为数据或资料(我国的台湾和香港称数据为“资料”)。但是,数据也只是一些符号,只有解释以后,才成为信息。这样,就能区分数据和信息了:信息强调含义(meaning),数据强调载体(media);信息强调实际效用,数据强调客观形式。两者之间联系的桥梁是“解释”。而解释是人类的一种智力活动。人的认识因背景不同,会对于同一数据作出不同的解释。例如,写出“28”两个符号,有的人将它解释为自己的年龄,有的人可能会把它解释为自己口袋中的钱数,有的人还会将它解释为自己家的门牌号……另外,同一种信息可以有不同的表示形式,例如一个人的年龄,可以用语音表达,可以用不同的语言表达,也可以用一些实物的数量表达等。



图 1.1 萨师煊教授

2. 数据模型的概念

人们对于模型的一般概念，特别是具体的模型其实并不陌生，比如常常会听到类似“哇，这个坦克模型太逼真了！”的说法。这里说明两个问题，首先看到的不是真实的坦克，其次它又真实地反映了实际坦克的特征。这说明了模型不是真实的事物，但是模型却又能真实反映事物的特征。对现实世界事物特征的模拟和抽象就是这个事物的模型。所谓抽象就是去掉与问题的解决关系不大的枝节，只考虑对解决问题影响很大的方面，以便使问题容易理解、容易解决。这是人们解决复杂问题的一种基本方法。同样，处理数据的有效方法是建立数据模型。

数据是对客观世界中事物的描述。因此，数据处理时，要求人们首先理清现实世界中客观事物之间的关系，为此需要建立数据的概念模型。其次要让计算机能够方便地处理这些数据，为此要建立数据的逻辑模型及物理模型。图 1.2 描述了这个过程。

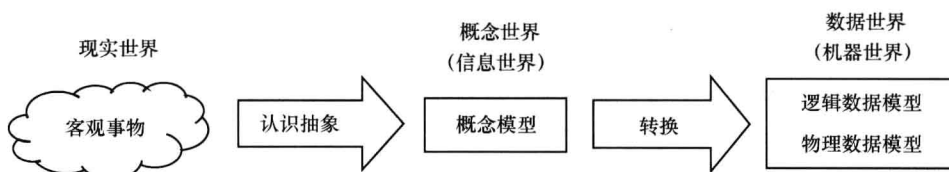


图 1.2 对现实世界事物抽象转换的 3 个阶段

1.1.2 数据的概念模型

1. 概念模型相关术语

概念模型也称信息模型，是为人们理清客观事物之间的关系而建立的模型，也是数据库设计人员与用户进行交流的工具。概念模型中的常用术语主要如下。

(1) 实体 (Entity): 现实世界存在的并可以相互区分的事物。实体可以是具体的事物，如学生、员工、商品、学校、企业等，也可以是抽象的事件，如比赛的赛况、货物的运输记录等。还可以指事物与事物之间的联系，如学生选课、顾客订票等。

(2) 属性 (Attribute): 实体具有的特征称为属性。如学生的学号、姓名、性别，商品的名称、单价等都是该实体具有的特征，也称为该实体的属性。每个实体都可用若干属性进行刻画，选取的实体属性越多，刻画出的实体就越清晰。属性有“属性型”和“属性值”的概念，属性的名称及其取值数据类型称为属性型，属性所取的具体值就是属性值。如学生实体中有姓名属性，那么“姓名”和取值字符类型就是属性型，而“张三”则是属性值。

(3) 域 (Domain): 属性的取值范围称为该属性的域。如性别属性的域是“男”或“女”，期末考试成绩的域是 0~100。

(4) 码 (Key): 在实体的众多属性中能够相互区分每一个（唯一标识）实体的属性或属性的组合称为实体的码。如学生的学号可以作为学生实体的码。

(5) 实体集 (Entity Set): 性质相同的同类实体的集合称为实体集，如一个学校的所有学生的集合、一个超市所有商品的集合等都可以称作实体集。

(6) 联系 (Relationship): 实体之间的对应关系称为实体间的联系。

2. 概念模型的表示方法——E-R 图

数据库设计的重要任务之一就是要建立概念数据库的具体描述，即概念模型。描述概念模型的主要工具是实体-联系模型，也称为 E-R 模型 (Entity-Relationship 模型) 或 E-R 图。用 E-R 模型描述的概念模型是独立于具体的数据库管理软件所支持的数据模型的，它是各种数据模型的共

同基础。

E-R 图主要由实体、属性和联系 3 个要素组成,并分别使用了表 1.1 中所描述的几种基本符号进行表示。

表 1.1

E-R 模型中的图形符号

图形符号	含 义
	矩形表示实体,实体名写在框内
	椭圆表示实体的属性,属性名写在圆内
	菱形表示实体间的联系,联系名写在框内
—	无向连接上述 3 种图形,构成完整 E-R 模型

图 1.3 所示为一个教师实体的属性关系。图 1.4 所示为图书借阅系统中的 E-R 图,该图描述了借阅者和图书两个实体之间存在着多对多的联系,说明了一本图书在不同的时间可以借给多个借阅者,且一个借阅者可以同时借阅多本图书。



图 1.3 教师实体图

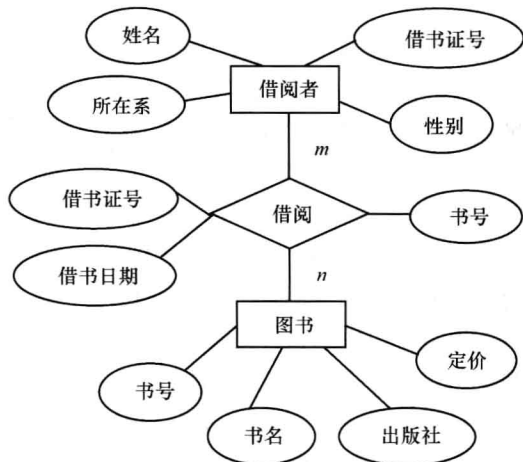


图 1.4 图书借阅 E-R 图

3. 实体间联系的种类

实体间的联系具体指一个实体集中的每一个实体与另外一个实体集中实体存在的联系。它反映的是现实世界事物与事物之间的相互联系。实体之间的联系归纳起来主要有以下 3 种类型。

(1) 一对一联系 (1:1): 如果对于某实体集 A 中的每一个实体,在另一实体集 B 中有且仅有一个实体与之联系,反之亦然,则称实体集 A 和实体集 B 之间存在着一对一的联系,并记作 1:1。例如学校和校长之间的联系就是一对一的联系,因为每一所学校只有一个校长,每一个校长只在一所学校任职。学校与校长之间的一对一联系表示如图 1.5 所示。



图 1.5 一对一联系

(2) 一对多联系 (1:n): 如果对于某实体集 A 中的每一个实体,在另一实体集 B 中有多个实

体与之联系,反之,对于实体集 B 中的每一个实体,在实体集 A 中都只有一个实体与之联系,那么就称实体集 A 与实体集 B 之间存在着一对多的联系,或者也可以说实体集 B 与实体集 A 之间存在着多对一的联系。并记作 $1:n$ 或 $n:1$ 。例如班级和学生之间的联系就是一对多的,因为每一个班级有很多个学生,而每一个学生只属于一个班级。班级与学生之间的一对多联系表示如图 1.6 所示。

(3) 多对多联系 ($n:m$): 如果对于某实体集 A 中的每一个实体,在另一实体集 B 中有多个实体与之联系,反之,对于实体集 B 中的每一个实体,在实体集 A 中也有多个实体与之联系,那么就称实体集 A 与实体集 B 之间存在着多对多的联系。并记作 $n:m$ 。例如,学生和课程之间的联系就是多对多的,因为每一个学生要选修多门课程,且每一门课程又有多个学生选修。学生与课程之间的多对多联系表示如图 1.7 所示。



图 1.6 一对多联系



图 1.7 多对多联系

1.1.3 数据的逻辑模型

数据的逻辑模型是指数据在计算机系统中被处理时的组织结构。人们常说的数据模型即指逻辑数据模型。迄今为止,人们已经使用的数据逻辑模型有 3 种:层次模型、网状模型和关系模型。逻辑模型不同,描述和实现方法也不同,相应的支持软件也不同。

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。层次数据库系统采用层次模型作为数据的组织方式,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,这是 1968 年 IBM 公司推出的第一个大型的商用数据库管理系统,曾经得到广泛的使用。

层次模型是以每个实体为结点,用树形结构来表示实体及其之间的联系。在层次模型中,数据被组织成由“根”开始的一棵倒置的“树”,连线(有向边)始端结点叫做父(双亲)结点,下一层次结点叫做子(孩子)结点。树中的每一个结点代表实体型,连线则表示实体间的联系。图 1.8 所示为层次模型的示意图,图 1.9 所示为层次模型的一个实例。

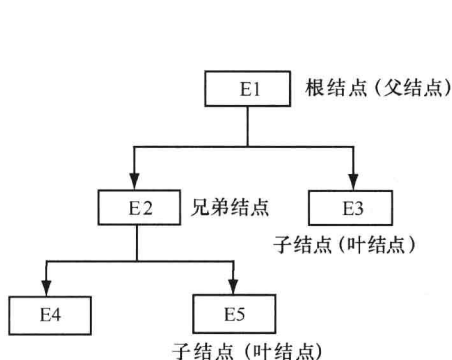


图 1.8 层次模型示意图

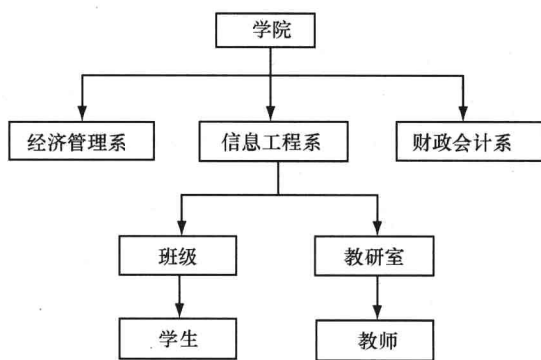


图 1.9 校务系统层次模型

根据树形结构的特点,层次模型的主要特征如下:

- (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) 根结点以外的其他结点有且只有一个双亲结点,而向下可以有若干结点。

层次模型具有层次清晰、结构简单、易于实现等优点，且由于受到上述两个条件限制，它可以比较方便地表示一对多和一对一的实体联系，但不能直接表示多对多的实体联系，对于复杂的数据库关系，层次模型实现起来存在局限性。

2. 网状模型

网状模型用以实体为结点的有向图来表示各实体及其之间的联系。网状模型是一种比层次模型更加具有普遍性的结构，它忽略了层次模型的两个限制条件，而且允许任意两个结点之间不受层次的限制实现多种联系。因此可以说，层次模型是网状模型的一个特例。图 1.10 所示为网状模型的一个实例。

网状模型虽然能够更加直观地描述现实世界，可以直接表示多对多的实体联系，且各实体间的联系是用指针实现的，具有较高的查询效率，但是其结构复杂，技术上的具体实现会比较困难，众多的指针也使得管理工作变得相当复杂，对用户而言掌握和使用也都比较麻烦。

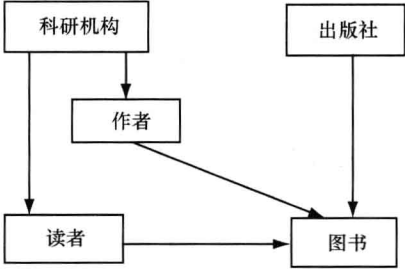


图 1.10 科研机构网状模型

3. 关系模型

1970 年，有“关系数据库之父”之称的 IBM 公司研究员埃德加·弗兰克·科德（Edgar Frank Codd）提出了数据库关系模型的理论，首次运用数学方法来研究数据结构和数据操作，将数据库设计从以经验为主提高到以理论为指导。

关系模型的结构与层次模型和网状模型相比有着本质的区别，它用二维表格来表示实体及其之间的联系。在关系模型中，每一张二维表就称为一个关系，描述了一个实体集，每一行在关系中称为元组（记录），每一列在关系中称为属性（字段）。每张二维表都有一个名称，也即为该关系的关系名。比如学生关系的每一行代表一个学生的记录，每一列代表学生的一个属性，如表 1.2 所示。

表 1.2 学生关系

学 号	姓 名	性 别	出生日期	专 业	所在系
20123040158	方芳	女	1993-1-10	网络工程	信息工程系
20123030101	陈瑜	男	1992-12-26	国际经济与贸易	经济管理系
20123010102	冯乔恩	女	1993-6-18	英语教育	外语系
20123020103	黄雯	女	1990-10-2	会计学	财政会计系

关系模型与其他模型相比，具有数据结构单一、理论严谨、简单明了、易学易用等特点，是目前最流行、最重要、使用最广泛的数据模型之一。

1.2 数据库与数据库系统

1.2.1 数据管理技术的产生与发展

数据管理是指如何对数据进行分类、编码、组织、存储、检索、维护等。例如，超市要对商

品的买卖进行记账、开具发票,学校教学管理部门要对学生、课程、成绩及教师等数据进行收集、存储、管理等。数据管理是数据处理的核心。伴随着计算机软、硬件技术的发展以及计算机应用的不断扩展,数据管理技术的水平也得到了前所未有的提高与发展,数据库技术已成为这个时代数据管理的重要基础和核心技术。也可以说,数据库技术的产生与发展是伴随着数据管理技术的不断发展而逐步形成的。数据管理技术从产生到发展大致经历了人工管理、文件系统管理、数据库系统管理 3 个阶段。

1. 人工管理阶段

在计算机技术出现之前,人们运用常规的手段从事数据的记录、存储和加工,如利用纸张来记录和利用计算工具(算盘、计算尺等)来进行计算,并主要使用人的大脑来管理和利用这些数据。20 世纪 50 年代计算机开始用于数据处理。但是,当时的数据处理水平极为原始,数据以变量或数组的形式存在,所能处理的数据量很小;数据无结构,数据间缺乏逻辑组织;数据依附于处理它的程序,缺乏独立性。以学校的数据管理为例,在此阶段,应用程序与数据的关系如图 1.11 所示。

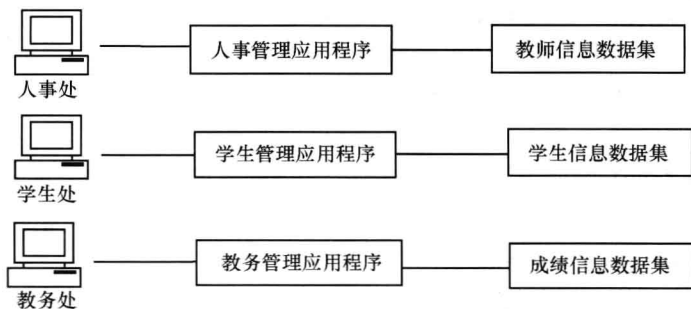


图 1.11 人工管理阶段应用程序与数据的关系

2. 文件系统阶段

20 世纪 50 年代后期到 60 年代中期,计算机的处理速度和存储能力惊人地提高,计算机开始大量用于数据管理,在硬件方面,出现了直接存取的大容量外存储器,如磁盘等,数据可以长期保存在计算机外存上;在软件方面,出现了操作系统,其中包含文件系统。文件系统为应用程序和数据之间提供了一个公共接口,使得应用程序可以通过统一的存取方法来存取、操作数据,程序和数据之间不再直接对应,因而具备了一定的独立性。这一阶段文件系统实现了记录内的结构化,但从文件的整体来看却仍是无结构的。

然而当数据量增加或数据的使用用户数量越来越多时,文件系统管理便难以适应数据管理的需求了,主要表现为以下几点:数据面向特定的应用程序,因此独立性较差;数据文件根据应用程序的需要而建立,因此造成数据的冗余度大、数据共享性差;缺乏对数据的统一控制和管理,各个数据文件没有统一的管理机构,因此管理和维护的代价也比较大。文件系统阶段学校数据管理中应用程序与数据的关系如图 1.12 所示。

3. 数据库系统阶段

20 世纪 60 年代后期,计算机性能得到进一步提高,更重要的是出现了大容量磁盘,存储容量大大增加且价格下降,克服了文件系统阶段管理数据时的不足,而解决实际应用中多个用户、多个应用程序共享数据的要求,从而使数据能为尽可能多的应用程序服务,这就出现了数据库这样的数据管理技术。这一阶段使用专门的数据库管理软件——数据库管理系统(DBMS)实现对数据进行统一的控制和管理。相关数据按统一的数据模型,以记录为单位,用文件方式存储在数据库中,具有整体的结构性,数据不再只针对某一个特定的应用程序,而是面向全组织,在应用程序和数据库

之间保持高度的独立性，数据共享性高，冗余度低，具有完整性、一致性、可靠性的特点。

数据库系统阶段学校数据管理中应用程序与数据的关系如图 1.13 所示。

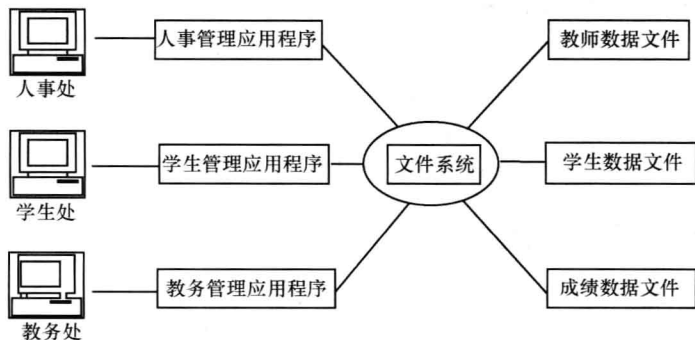


图 1.12 文件系统阶段应用程序与数据的关系

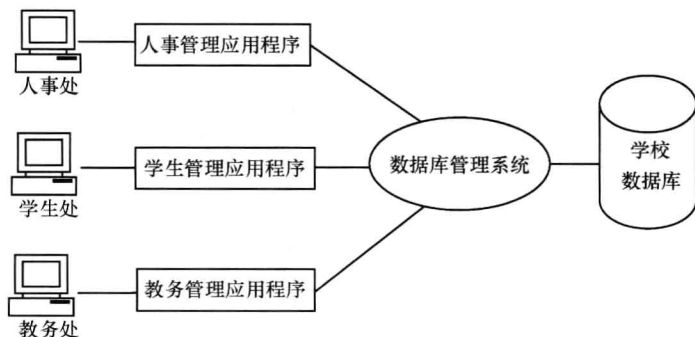


图 1.13 数据库系统阶段应用程序与数据的关系

1.2.2 数据库及其特点

数据库（DataBase, DB）就是存放数据的仓库。只不过这个仓库是依赖于计算机、按一定格式、可以长期存放数据的。例如，学校的教务管理数据库中有组织地存放了学生基本情况、课程情况、学生成绩情况、排课情况和教师情况等数据，相关部门可以借助数据库管理软件对这些数据进行综合管理，并且这些数据可供教务处、任课教师和学生等共同使用。

数据库技术的出现是计算机数据管理技术的重大进步，它具有以下主要特点。

1. 数据共享

数据共享允许多个用户或应用程序可以同时访问和使用同一数据库中数据而互不干扰，为多种程序设计语言提供编程接口。DBMS 提供并发和协调机制，保证多个应用程序进行数据共享时不会产生任何冲突，保证数据不遭到破坏。

2. 数据独立性

数据独立性指数据与应用程序之间的彼此独立，不相互依赖，数据存储结构的改变不影响使用数据应用程序的正常运行。数据独立性包括物理独立性和逻辑独立性。物理独立性指数据存储格式和组织方法的改变不影响数据库的逻辑结构，所以不影响应用程序；逻辑独立性指数据库逻辑结构的改变不影响用户的应用程序，即应用程序不需修改仍可继续正常运行。

3. 减少数据冗余

数据冗余是指一种数据存在多个相同的副本，即数据重复。数据冗余既浪费存储空间，又容

易产生数据的不一致。数据库从全局观念组织和存储数据, 库中的数据根据特定的数据模型结构化, 从而大大地减少了数据冗余, 增强了数据一致性, 提高了数据使用效率。

4. 数据安全性和完整性保护增强

数据库系统可以提供一系列有效的安全保密机制, 阻止未授权用户非法进入系统或访问数据。DBMS 提供数据完整性检查机制, 避免不合法、不正确、不一致、无效的数据进入数据库中, 保证数据库的数据完整性。另一方面, 数据库系统还提供了一系列的数据备份与恢复措施, 确保当数据库遭到破坏时也可及时恢复数据, 以减少损失。

1.2.3 数据库管理系统与数据库应用系统

1. 数据库管理系统

任何仓库都有管理机构。数据库也有管理机构, 就是数据库管理系统 (Data Base Management System, DBMS)。它是位于用户与操作系统之间的一种帮助用户实现对数据库的定义、建立、操作、管理和维护的数据库管理软件。各种数据库命令的执行及访问数据库的操作都要通过数据库管理系统来统一管理和控制。其主要功能可以概括为以下几个方面。

(1) 数据定义功能: DBMS 提供的数据库定义语言 (Data Definition Language, DDL) 可以定义构成数据库结构的外模式、模式和内模式, 定义各个模式之间的映射, 定义数据的完整性约束等。

(2) 数据操纵功能: DBMS 提供的数据库操纵语言 (Data Manipulation Language, DML) 可以实现对数据库中数据的检索、插入、修改和删除等基本操作。

(3) 数据库运行管理: 包括对数据库进行并发控制、安全性检查、完整性约束条件的检查和执行、数据库的内部维护等。

(4) 数据组织、存储和管理: DBMS 负责分门别类地组织、存储和管理数据库中需要存放的各种数据, 确定以何种文件结构和存取方式合理地组织这些数据。

(5) 数据库的建立和维护: 主要包括通过 DBMS 完成对数据库的定义、创建及维护等操作。

(6) 数据通信功能: 提供与其他系统进行通信的功能。例如, 在分布式数据库或具备网络操作功能的数据库中提供数据库的通信功能, 又或者提供与其他 DBMS 或文件系统的接口。

数据库管理系统是整个数据库系统的核心, 是数据库系统的重要组成部分之一。

2. 数据库应用系统

数据库应用系统简称数据库系统 (Data Base System, DBS), 是指以计算机系统为基础, 以数据库方式管理大量共享数据的计算机应用系统。它一般由计算机系统、数据库、数据库管理系统、应用系统及相关的人员组成。计算机系统是数据库系统的基础。应用系统是在数据库管理系统基础上建立的面向不同应用的系统。例如教务管理系统、图书管理系统、铁路售票系统、网上销售系统等都是具备不同目标的应用系统。相关的人员主要包括负责管理与维护数据库的数据库管理员 (DBA) 以及数据库的最终用户。数据库系统的组成结构如图 1.14 所示。

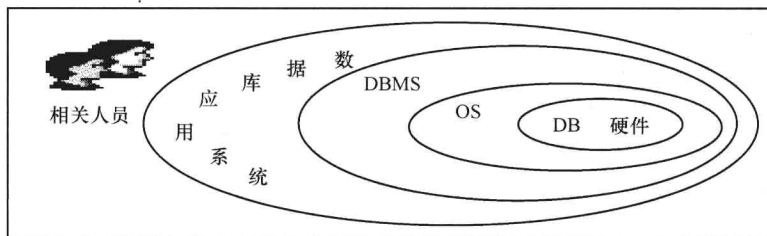


图 1.14 数据库系统的组成结构

1.3 关系数据库

自 20 世纪 80 年代以来,新推出的数据库管理系统几乎都支持关系模型,本书后面将要介绍的 Access 就是一种典型的关系数据库管理系统。

1.3.1 关系数据库概述

在数据库中,数据结构如果依照关系模型定义,就是关系数据库。关系数据库由许多不同的关系及其之间的联系构成,其中每个关系就是一张二维表,故关系数据也称为数据表。数据分别存储在各个数据表中,每张表包含某个特定主题的数据。

1. 关系术语

关系数据库中常用的关系术语主要如下。

(1) 关系 (Relation): 一张二维表就是一个关系。通常将一个没有重复行、重复列的二维表看成一个关系,每个关系都有一个关系名。例如,“学生表”、“图书表”、“商品表”等。

(2) 元组: 表中的一行叫做一个元组,元组对应表中具体的一条记录,描述的是现实世界中的一个实体。例如,“学生表”中一个学生的记录就是一个元组。

(3) 属性 (Attribute): 表中的一列叫做一个属性,也称为一个字段。例如,“学生表”中的“学号”、“姓名”等都是属性或字段。

(4) 域 (Domain): 属性的取值范围叫做域。例如,“学生表”中的“性别”字段的取值范围只能是“男”或“女”,这就是“性别”属性的域。

(5) 关键字 (码 (Key)): 关系中能唯一区分、确定不同元组的属性或属性组合,称为该关系的一个关键字 (或候选关键字)。例如,“学生表”中的“学号”、“身份证号”属性等都可以作为候选关键字。

(6) 主关键字 (主键 (Primary Key)): 在候选关键字中选定一个作为关键字,被选出的这个关键字就称为该关系的主关键字,或简称主键 (Primary Key)。例如,在“学生表”中我们一般选择“学号”作为该关系的主键。

(7) 外部关键字 (外键 (Foreign Key)): 如果关系 R 中的某个属性 (或属性组合) 不是 R 的主关键字,但它却是另一个关系的主关键字,则称此属性 (或属性组合) 为关系 R 的外部关键字,或简称外键 (Foreign Key)。在关系数据库中,主要用外部关键字来表示两个表间的联系。

(8) 关系模式: 对一个关系的描述称为一个关系模式,通常记作: 关系名 (属性 1, 属性 2, …, 属性 n)。例如,学生 (学号, 姓名, 性别, 专业, 系别, 年级, 籍贯)。

2. 关系的性质

关系数据库要求关系必须是规范的,每一个关系都必须具备如下 6 个基本条件。

- (1) 表中的每一数据项必须是原子的,不可再分。
- (2) 列是同质的,即每一列有相同的数据类型,且来自同一个域。
- (3) 不同的列 (属性) 可出自同一个域,但不同的属性必须要给予不同的属性名。
- (4) 列的顺序无要求,即任意两列的次序可以交换。
- (5) 行的顺序无要求,即任意两个元组的次序可以交换。
- (6) 任意两个元组不能完全相同。