



高等教育计算机系列“十三五”规划教材

数据库技术与应用 Access

▶▶ 主编 牛莉 董峰





高等教育计算机系列“十三五”规划教材

数据库技术与应用 Access

◎主 编 牛 莉 董 峰

◎副主编 黄春花 付卫平 张美华



图书在版编目 (CIP) 数据

数据库技术与应用:Access / 牛莉, 董峰主编. —长沙 :
湖南教育出版社, 2019. 2

ISBN 978-7-5539-5715-9

I. ①数… II. ①牛… ②董… III. ①关系数据库系统
—高等学校—教材 IV. ①TP311.132.3

中国版本图书馆 CIP 数据核字 (2019) 第 028040 号

策划编辑 晏密英

责任编辑 刘文婷

封面设计 袁敏

排版设计 邹满辉

书 名: 数据库技术与应用 Access
SHUJUKU JISHU YU YINGYONG Access
主 编: 牛莉 董峰
出版发行: 湖南教育出版社
地 址: 长沙市韶山北路 443 号
网 址: www.bakclass.com
发 行 部: 0731-85520531
编 辑 部: 0731-85558352 c sgaojiao@163.com
印 刷: 湖南华商文化商务有限公司
经 销: 湖南天易创图文化有限公司

开 本: 787mm×1092mm 1/16
印 张: 14.75
字 数: 340 千字
版 次: 2019 年 2 月第 1 版
印 次: 2019 年 2 月第 1 次印刷

书 号: ISBN 978-7-5539-5715-9
定 价: 48.00 元

本书如有印刷、装订错误, 可向承印厂调换

内容简介

本书是根据教育部高等教育司组织制订的《普通高等院校计算机基础课程教学大纲》、教育部全国计算机等级考试中心制订的《全国计算机等级考试二级 Access 数据库程序设计考试大纲（2016 年版）》、教育部计算机科学与技术教学指导委员会提出的《关于进一步加强高等学校计算机基础教学的意见》（简称“白皮书”）精神编写的。本书介绍了数据库基础知识、Access 基本知识 with 基本操作、VBA 编程等知识。

本书可作为高等院校本、专科学生的教学用书，也可作为学习数据库技术与应用的读者自学用书。

前 言

本书是根据教育部高等教育司组织制订的《普通高等院校计算机基础课程教学大纲》、教育部全国计算机等级考试中心制订的《全国计算机等级考试二级 Access 数据库程序设计考试大纲（2016 年版）》、教育部计算机科学与技术教学指导委员会提出的《关于进一步加强高等学校计算机基础教学的意见》（简称“白皮书”）精神编写的。

本书从与关系数据库管理系统相关的一些基础理论和概念讲起，引领读者了解 Access 的基本性能，介绍 Access 2010 的操作方法，通过大量实例讲解操作步骤，由浅入深，简单易懂，讲述了 VBA 数据库编程基础知识，让读者对面向对象程序设计方法有所了解。

全书共 7 章，第 1 章介绍数据库基础知识、数据模型和 Access 基础知识，第 2 章介绍数据库和数据表的创建、修改等基本操作，第 3 章介绍各种查询的相关操作，第 4 章介绍窗体的各种操作，第 5 章介绍各种报表的设计操作，第 6 章讲述了宏的相关知识，第 7 章对 VBA 编程作了较为详细的介绍。

本书可作为高等院校本、专科学生的教科书，也可作为学习数据库技术与应用的读者的自学用书。

本书由牛莉、董峰主编，黄春花、付卫平、张美华为副主编，牛莉负责全书的修改与统稿，张美华为本书提供了不少修改意见和建议。第 1~2 章由黄春花编写，第 3~6 章及附录由牛莉编写，第 7 章由董峰、付卫平编写。

在本书的编写过程中，我们得到学校有关领导和老师们的大力支持和帮助，在此深表感谢！由于作者水平有限，书中难免存在错误或不足之处，欢迎广大读者批评指正。

编 者

2018 年 7 月

目 录

第 1 章 数据库基础	1
1.1 数据管理发展概述	1
1.2 数据库系统概述	4
1.3 数据模型	7
1.4 关系数据库	10
1.5 ACCESS 概述	12
本章小结	18
思考题	18
第 2 章 数据库和数据表	19
2.1 数据库的操作	19
2.2 数据表的结构设计	25
2.3 数据表的创建	27
2.4 表之间关系的创建	37
2.5 表的基本操作	40
2.6 外部数据	49
本章小结	51
思考题	51
第 3 章 查询	52
3.1 查询概述	52
3.2 创建选择查询	53
3.3 设置查询条件	57
3.4 设置查询的计算	60
3.5 创建交叉表查询	62
3.6 创建参数查询	64
3.7 创建操作查询	65
3.8 创建 SQL 查询	69
本章小结	77
思考题	77
第 4 章 窗体	78

4.1	窗体概述	78
4.2	创建窗体	81
4.3	设计窗体	88
4.4	创建控制窗体	111
4.5	格式化窗体	116
	本章小结	120
	思考题	120
第 5 章	报表	121
5.1	报表概述	121
5.2	报表的创建	124
5.3	使用报表的计算控件	129
5.4	创建子报表	137
5.5	格式化报表	141
5.6	报表的预览和打印	145
	本章小结	147
	思考题	147
第 6 章	宏	148
6.1	宏概述	148
6.2	宏的创建及运行	149
6.3	常用宏操作	157
6.4	使用宏创建菜单	158
6.5	宏的应用	162
	本章小结	164
	思考题	164
第 7 章	VBA 数据库编程	165
7.1	面向对象程序设计基础概念	165
7.2	VBA 的模块及编程环境	167
7.3	VBA 编程基础	172
7.4	VBA 程序中的语句	181
7.5	VBA 程序中过程的调用和参数的传递	196
7.6	VBA 程序的错误处理和调试	199
7.7	VBA 编程访问数据库	202
	本章小结	208
	思考题	208

附录	209
附录一 字段、表与查询常用属性	209
附录二 窗体、报表与控制件常用属性	211
附录三 常用宏操作命令	216
附录四 常用函数	221
参考文献	226

数据库是在 20 世纪 60 年代末发展起来的一项重要技术，20 世纪 70 年代以来数据库技术得到迅猛发展，已经成为计算机科学与技术的一个重要分支。数据库技术和系统已经成为信息基础设施的核心技术和重要基础，数据库技术作为数据管理的最有效的手段，极大地促进了计算机应用的发展。数据库技术主要是用来解决数据处理的非数值计算问题，数据处理的主要内容是数据的存储、查询、修改、排序和统计等。本章主要介绍数据库、数据库系统、数据库管理系统、关系模型及关系数据库基本理论和 Access 系统初步知识。

1.1 数据管理发展概述

计算机数据管理技术随着计算机硬件、软件和计算机应用的发展而发展，已经历了从较为低级的人工管理到先进的数据库、数据仓库、数据挖掘的演变。

1.1.1 数据与数据处理

数据 (data) 是指存储在某种介质上能够识别的物理符号，如图形符号、数字、字母等，是信息的表现形式。数据在实际应用中，有两种基本形式：一种是数值型，即可以参与数值运算的数据，如表示成绩、工资的数据；另一种是非数值型，即由字符组成、不能参与数值运算的字符型数据。信息是客观世界的抽象描述，是数据所包含的意义；信息是加工处理后的数据，是数据所表达的内容，是有用的数据。

数据处理是指将数据转换成信息的过程。现实世界中的数据往往是原始的、非规范的，但它是数据的原始集合，对这些原始数据进行处理，才能产生新的数据 (信息)。这一处理包括对数据的收集、记录、分类、排序、存储、计算/加工、传输、制表和递交等操作。从数据处理的角度而言，信息是一种被加工成特定形式的数据，这种数据形式对于数据接收者来说是有意义的。因此，人们时常说起的“信息处理”，其真正含义应该是为了产生信息而处理数据。

1.1.2 人工管理

在 20 世纪 50 年代中期以前，计算机主要用于科学计算，计算机上没有操作系统，没有管理数据的专门软件，也没有像磁盘这样的设备来存储数据。这个时期数据管理有如下特点：

- (1) 数据不长期保存，程序运行结束后就退出计算机系统。
- (2) 数据和程序一一对应，即一组数据对应一个程序。不同应用程序的数据之间是相互独立、彼此无关的。
- (3) 没有软件系统对数据进行管理，程序员不仅要规定数据的逻辑结构，还要在程序中设计物理结构，包括存储结构、存取方法及输入输出方式等。也就是说，数据对程序不具有独立性，数据是程序的组成部分，一旦数据在存储上有所改变，必须修改程序。

1.1.3 文件系统

从 20 世纪 50 年代后期至 60 年代中期, 数据管理处于文件系统阶段。文件系统提供了在外存储器上长期保存数据并对数据进行存取的手段。程序与数据有一定的独立性。

应用程序利用操作系统的文件管理功能, 将相关数据按一定的规则构成文件, 通过文件系统对文件中的数据进行存取、管理, 实现数据的文件管理方式。

文件系统为程序与数据之间提供了一个公共接口, 使应用程序采用统一的存取方法来存取、操作数据, 程序与数据之间不再是直接的对应关系, 因而程序和数据有了一定的独立性。

文件系统数据管理有如下缺点:

- (1) 数据冗余度大, 极易造成数据的不一致。
- (2) 缺乏数据独立性, 程序与数据相互依赖。
- (3) 数据无集中管理, 安全性与完整性无法保证。

文件系统存在的问题阻碍了数据处理技术的发展, 不能满足日益增长的信息需求, 这正是数据库技术产生的原动力, 也是数据库系统产生的背景。

1.1.4 数据库系统

从 20 世纪 60 年代后期开始, 需要计算机管理的数据量急剧增长, 并且对数据共享的需求日益增强。文件系统的数据库管理方法已无法适应开发应用系统的需要。为了实现计算机对数据的统一管理, 达到数据共享的目的, 人们发展了数据库技术。

数据库技术的主要目的是有效地管理和存取大量的数据资源, 包括提高数据的共享性, 使多个用户能够同时访问数据库中的数据; 减少数据的冗余度, 提高数据的一致性和完整性; 提供数据与应用程序的独立性, 从而减少应用程序的开发和维护代价。

为数据库的建立、使用和维护而配置的软件成为数据库管理系统 DBMS (Database Management System)。数据库管理系统利用了操作系统提供的输入/输出控制和文件访问功能, 因此它需要在操作系统的支持下运行。数据库系统中, 数据与程序的关系如图 1-1 所示。

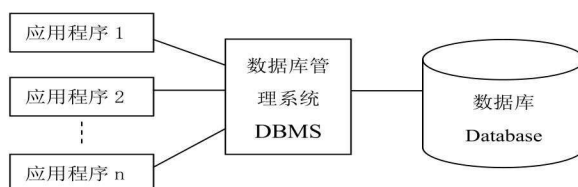


图 1-1 数据库系统中数据与程序

1.1.5 分布式数据库系统

分布式数据库系统是在集中式数据库系统的基础上发展起来的, 是计算机技术和网络技术结合的产物。随着传统的数据库技术日趋成熟, 计算机网络技术的飞速发展和应用范围的扩充, 数据库应用已经普遍建立在计算机网络之上, 这时集中式数据库系统表现出它的不足之处: 数据按实际需要已在网络上分布存储, 再采用集中式处理, 势必会出现通信开销大的情况; 应用程序集中在一台计算机上运行, 一旦该计算机发生故障, 则整个系统都会受到影响, 可靠性不高; 集中式处理导致系统的规模和配置都不够灵活, 系统的可扩展性差。在这种形势下, 集中式数据库的“集中计算”开始向“分布计算”发展。

分布式数据库系统有两种: 一种在物理上是分布的, 但在逻辑上却是集中的; 另一种在物理上和逻辑上都是分布的, 也就是联邦式分布数据库系统。

1.1.6 面向对象数据库系统

20世纪90年代以来,在关系型数据库的基础上,引入面向对象技术,从而使关系型数据库发展成为一种新型的面向对象关系型数据库。面向对象的程序设计方法是目前程序设计中的主要方法之一,它简单、直观、自然,十分接近人类分析和处理问题的自然思维方式,同时又能够有效地用来组织和管理不同类型的数据。

面向对象数据库系统是面向对象的程序设计技术与数据库技术相结合的产物。它的主要特点是具有面向对象技术的封装性和继承性,提高了软件的可重用性。

1.1.7 数据仓库

随着客户机/服务器技术的成熟和并行数据库的发展,信息处理技术实现了从大量的事务型数据库中抽取数据,并将其清理、转换为新的存储格式的过程,即为决策目标把数据聚合在一种特殊的格式中。随着此过程的发展和完善,这种支持决策的、特殊的数据即为数据仓库。

数据仓库就是面向主题的、集成的、随时间变化的,但信息本身相对稳定的数据集合,它用于对管理决策过程的支持。数据仓库中的数据面向主题,与传统数据库面向应用相对应。

1.1.8 数据挖掘

数据挖掘,又称为数据库中的知识发现,就是一个从数据库中获取有效的、新颖的、潜在有用的、最终可理解的知识的复杂过程。简单来说,数据挖掘就是从大量数据中提取或“挖掘”知识。

数据挖掘和数据仓库的协同工作,一方面可以迎合和简化数据挖掘过程中的重要步骤,提高数据挖掘的效率和能力,确保数据挖掘过程中数据来源的广泛性和完整性;另一方面,数据挖掘技术已经成为应用中极为重要和相对独立的工具。

1.1.9 大数据

“大数据”(big data),或称巨量数据、海量数据、大资料,指无法在一定时间范围内用常规软件工具进行捕捉、管理和处理的数据集合,是需要新处理模式才能具有更强的决策力、洞察发现力和流程优化能力的海量、高增长率和多样化的信息资产。一般认为,大数据具有“4V”特点:第一,数量(volume),即数据巨大,从TB级别跃升到PB级别(1PB=1024TB);或从PB级别跃升到EB级别(1EB=1024PB)。第二,多样性(variety),即数据类型繁多,不仅包括传统的格式化数据,还包括来自互联网的网络日志、视频、图片、地理位置信息等。第三,速度(velocity),即处理速度快,1秒定律(在秒级时间范围内给出分析结果,超出这个时间,数据就失去价值了),可从各种类型的数据中快速获得高价值的信息。第四,价值性(value),大数据最大的价值在于从大量不相关的各种类型的数据中,挖掘出对未来趋势与模式预测分析有价值的信息,并通过机器学习方法、人工智能方法或数据挖掘方法深度分析,发现新规律和新知识,运用于农业、金融、医疗等各个领域,从而最终达到改善社会治理、提高生产效率、推进科学研究的效果。

对于大数据而言,随着云计算、分布式处理技术、存储技术和感知技术等技术的发展,这些原本很难收集和使用的数据开始变得容易利用起来了。通过各行各业的不断创新,大数据必将为人类创造更多的价值。

1.2 数据库系统概述

1.2.1 基本概念

数据库系统 (Database System, DBS) 是由计算机系统、数据库及其描述机构、数据库管理系统和有关人员组成的具有高度组织性的总体。

1. 数据库的组成

一般来说, 数据库系统由五部分组成: 计算机系统、数据库、数据库管理系统、数据库管理员和用户。它们之间的层次如图 1-2 所示。

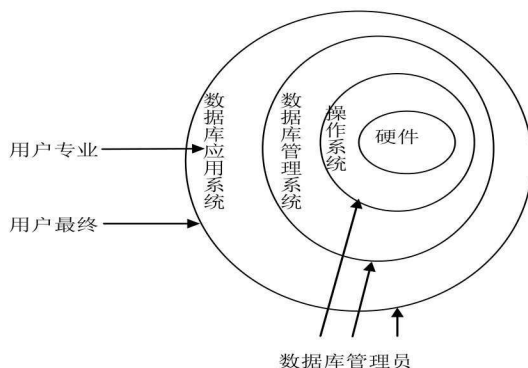


图 1-2 数据库系统层次示意图

(1) 计算机系统 (Computer System)。计算机系统是指提供数据库系统运行的硬、软件平台。硬件平台一般包括计算机中央处理器、足够大的内存、足够大容量的磁盘等联机直接存取设备和较高通道能力, 以支持对外存的频繁访问, 同时还包括足够多的联机存储介质。软件平台指计算机操作系统提供的运行环境及开发工具。计算机系统提供的运行环境不恰当, 将会影响数据库系统的运行效率或者导致其根本无法运行。

(2) 数据库 (Database, DB)。数据库是指存储在计算机外存设备或网络存储设备上的结构化的相关数据集合。它不仅包括描述事物的数据本身, 还包括相关事物之间的联系。

数据库中的数据面向多种应用, 可以被多个用户、多个应用程序共享。例如, 某企业、组织或行业所涉及的全部数据的汇集。其数据结构独立于使用数据的程序, 对于数据的增加、删除、修改和检索由系统软件统一控制。

(3) 数据库管理系统 (DBMS)。为了让多种应用程序并发地使用数据库中具有最小冗余度的共享数据, 必须使数据与程序具有较高的独立性。这就需要有一个软件系统对数据实行专门管理, 即数据库管理系统。数据库管理系统是管理数据库的工具, 是应用程序与数据库之间的接口, 是为数据库的建立、使用和维护而配置的软件。

它是数据库系统的核心, 建立在操作系统的基础上, 实现对数据库的统一管理和控制。它需要解决科学地组织和存储数据、高效地获取和维护数据这两个问题。

它的主要功能包括以下四个方面:

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数据操作功能: DBMS 提供数据操作语言 (Data Manipulation Language, DML), 用户可以使用 DML 操作数据实现对数据库的基本操作, 如查询、插入、删除和修改等。

③数据库的运行管理功能: 包括并发控制、存取控制 (安全性检查)、完整性约束条件的检查与执行等。所有数据库的操作都要在这些控制程序的统一管理下进行, 以保证事务的正确运行和数据库数据的正确有效。

④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, 数据库的重组功能和性能监视、分析功能等。

(4) 数据库管理员 (Database Administrator, DBA)。数据库管理员是负责建立、维护和管理数据库系统的操作人员, 他们有丰富的计算机应用经验, 对业务数据的性质、结构、流程有较全面的了解。DBA 的职责包括定义并存储数据库的内容、监督和控制数据库的使用、负责数据库的日常维护、必要时重新组织和改进数据库。

(5) 用户 (User)。数据库系统的用户分为两类。一类是最终用户, 主要对数据库进行联机查询或通过数据库应用系统提供的界面来使用数据库, 如操作员、企业管理人员、工程技术人员, 他们不必了解数据库系统的结构和模式。另一类为专业用户 (即程序员), 他们负责设计应用系统的程序模块, 对数据库进行操作。他们有更多的计算机专业知识, 可对所授权使用到的数据库 (或视图) 进行查询、插入、删除、修改操作, 因此他们要了解数据库的外模式。

2. 数据库系统的特点

(1) 实现数据共享。在数据库系统中, 对数据的定义和描述已经从应用程序中分离出来, 而是通过数据库管理系统来统一管理。数据库中的数据不仅可以被同一企业或结构内的各个部门共享, 也可以被不同单位、地域甚至不同国家的用户共享。

(2) 采用特定的数据模型。数据库中的数据是有结构的, 这种结构由数据库管理系统所支持的数据模型表现出来, 任何数据库管理系统都支持一种抽象的数据模型。数据库中的数据文件是有联系的, 在整体上服从一定的结构形式。

(3) 具有较高的数据独立性。数据独立性是指数据独立于应用程序而存在。在文件系统中, 数据结构和应用程序相互依赖、相互影响。数据库系统则力求减少这种依赖, 实现数据的独立性。在数据库系统中, 数据库管理系统提供映像功能, 实现了应用程序对数据的总体逻辑结构、物理存储结构之间较高的独立性。用户只以简单的逻辑结构来操作数据, 无须考虑数据在存储器上的物理位置与结构。

(4) 减少了数据冗余。文件系统中数据专用, 每个用户拥有和使用自己的数据, 造成许多数据重复, 这就是数据冗余。在数据库系统实现共享后, 不必要的重复数据将被删除, 但为了提高查询效率, 有时也保留少量重复数据, 其冗余度可由设计人员控制。

(5) 有统一的数据控制功能。为保证多个用户能同时正常地使用同一个数据库, 数据库系统提供以下三方面的数据控制功能:

① 安全性控制: 数据库设置一套安全保护措施, 保证只有合法用户才能进行指定权限的操作, 防止非法使用所造成的数据泄密和破坏。

② 完整性控制: 数据库系统提供必要措施来保证数据的正确性、有效性和相容性, 当计算机系统出现故障时, 提供将数据恢复到正确状态的相应机制。

③ 并发控制: 当多用户并发进程同时存取、修改数据库时, 可能会发生相互干扰, 使数据库的完整性遭到破坏。因此, 数据库系统提供了对并发操作的控制功能, 对多用户的并发操作予以控制和协调, 保证多个用户的操作不相互干扰。

1.2.2 数据库系统的三级模式和二级映射结构

1972 年, 美国国家标准协会计算机与信息处理委员会 ANSI/X3 成立了一个 DBMS 研究组, 试图规定一个标准化的数据库系统结构, 规定总体结构、标准化数据库系统的特征, 包括数据库系统的接口和各部分所提供的功能, 这就是有名的 SPARC (Standard Planning and Requirement Committee) 分级结构。这种结构以内模式、概念模式和外模式三个层次来描述数据库。

数据库系统的体系结构是数据库系统的一个总的框架, 尽管实际的数据库系统的软件产品名目繁多, 如支持不同的数据模型、使用不同的数据库语言、建立在不同的操作系统环境之上、各有不同的存储结构等, 但数据库系统在总的体系结构上都具有三级模式的结构特征。

它们之间的联系经过两次转换, 把用户所看到的数据变成计算机存储的数据, 即三级模式、两级映像。如图 1-3 所示。

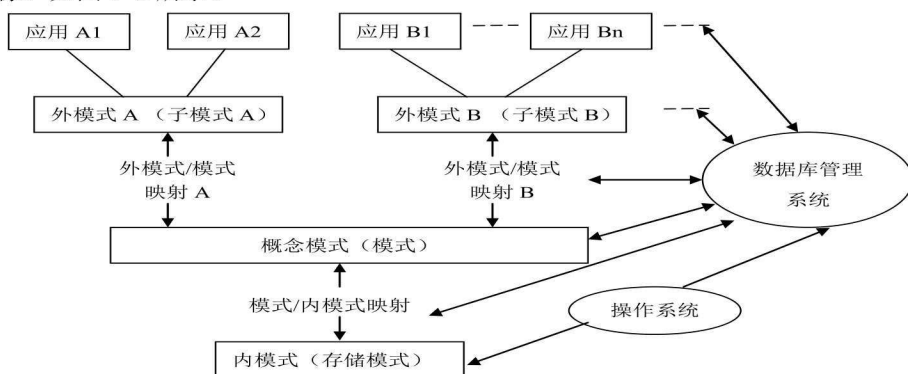


图 1-3 数据库的三级体系结构

1. 三级模式

(1) 外模式。外模式也称子模式或用户模式, 它是用户 (包括应用程序员和最终用户) 看见和使用的局部数据的逻辑结构和特征的描述, 是用户的数据视图, 是与某一应用有关的数据的逻辑表示。一个应用只能启动一个外模式, 一个外模式可以为多个应用启用, 如图 1-3 中外模式 A 被应用 A1 和应用 A2 启用。对于同一个对象, 因不同的需求, 使用不同的程序设计语言, 不同的用户外模式的描述可能各不相同, 同一数据在外模式中的结构、类型、长度、保密级别都可能不同。

外模式属于模式的子集。数据库系统提供外模式数据定义语言外模式 DDL, 用外模式 DDL 写出的一个用户数据视图的逻辑定义的全部语句称为此用户的外模式。

(2) 概念模式。概念模式简称为模式, 是数据库中全体数据的逻辑结构和特征的描述, 是所有用户的公共数据视图。

概念模式不同于外模式, 它与具体的应用程序及高级语言无关, 也不同于内模式, 比内模式抽象, 不涉及数据的物理存储结构和硬件环境。数据库系统提供概念模式描述语言 (模式 DDL) 来严格地定义模式所包含的内容, 用模式 DDL 写出的一种数据库逻辑定义的全部语句, 称为数据库的模式。模式是对数据库结构的一种描述, 而不是数据库本身, 它是装配数据库的一个框架。

(3) 内模式。内模式又称为存储模式, 是全部数据在数据库系统内部的表示或底层描述, 即为数据的物理结构和存储方法的描述。

内模式具体描述了数据如何组织并存入外部存储器上, 一般由系统程序员根据计算机系统的软硬件配置决定数据存储方法, 并编制程序实现存取, 因而内模式对用户是透明的。内模式是用内模式描述语言 (内模式 DDL) 来描述或定义。

2. 二级映射

(1) 外模式/模式映射。它定义了某个外模式和模式之间的对应关系, 是数据的全局逻辑结构和数据的局部逻辑结构之间的映像, 这些映像定义通常包含在各自的外模式中。当系统要求改变模式时, 可改变外模式/模式的映射关系而保持外模式不变。如数据管理的范围扩大或某些管理的要求发生改变后, 数据的全局逻辑结构发生变化, 对不受该全局变化影响的局部而

言,最多改变外模式与模式之间的映像,基于这些局部逻辑结构所开发的应用程序就不必修改。这种特性称为用户数据的逻辑数据独立性。

(2) 模式/内模式映射。它定义了数据逻辑结构和存储结构之间的对应关系,当数据库的存储结构发生了改变,如存储数据库的硬件设备发生变化或存储方法变化,引起内模式的变化,由于模式和内模式之间的映像使数据的逻辑结构可以保持不变,因此应用程序可以不必修改。这种全局的逻辑数据独立于物理数据的特性称为物理数据独立性。

由于有了上述两种数据独立性,数据库系统就可将用户数据和物理数据结构完全分开,使用户避免烦琐的物理存储细节。由于用户程序不依赖于物理数据,也就减少了应用程序开发和维护的难度。

1.3 数据模型

数据库需要根据应用系统中数据的性质、内在联系,按照管理的要求来设计和组织。数据模型是从现实世界到机器世界的一个中间层次。现实世界的事物反映到人的大脑中,人们把这些事物抽象为一种既不依赖于具体的计算机系统又不依赖于具体的 DBMS 的概念模型,然后把该概念模型转换为计算机中某个 DBMS 所支持的数据模型。

数据模型包括概念数据模型、逻辑数据模型及物理数据模型。

1.3.1 概念数据模型

概念数据模型是面向数据库用户的现实世界的数据库模型,也简称概念模型。概念模型主要用来描述现实世界的概念化结构。它使数据库的设计人员在设计的初始阶段,摆脱计算机系统及 DBMS 的具体技术问题,集中精力分析数据以及数据之间的联系等。概念模型与具体的计算机平台和具体的 DBMS 无关。概念模型是整个数据模型的基础。在此,仅介绍最广泛使用的概念模型设计方法——E-R 方法,设计概念模型的有关内容。

1.概念模型中的基本概念

(1) 实体。客观存在并且可以相互区别的事物称为实体。实体可以是实在的事物,也可以是抽象事件。例如,商品、客户等属于实体,它们是实际事物,而销售、订购、比赛等活动也是实体,它们是比较抽象的事件。

(2) 属性。描述实体的特征称为属性。例如,学生实体用学号、姓名、性别、政治面貌等若干属性来描述。

(3) 实体集和实体型。属性值的集合表示一个实体,而属性的集合表示一种实体的类型,称为实体型。同类型的实体的集合称为实体集。例如,在学生实体集当中,(1140201001,许德力,男,山东)表示学生名册中的一个具体人。

在 Access 中,用“表”来存放同一类实体,即实体集。Access 中一个“表”包含若干个字段,表中所包含的“字段”就是实体的属性。字段值的集合组成表中的一条记录,代表一个具体的实体,即每一条记录表示一个实体。

2.实体间的联系及分类

实体之间的对应关系称为联系,它反映现实世界事物之间的相互关联。如一个学生可以有多个任课老师,一个任课老师可以教多个学生。实体间联系的分类型是指一个实体型中可能出现的每一个实体与另一个实体型中多少个具体实体存在联系。两个实体间的联系可以归结为以下三种类型。

(1) 一对一联系 (1:1): 实体集 A 中的每一个实体与实体集 B 中的一个实体对应, 反之亦然, 记为 1:1。例如, 学校和正校长两个实体型, 学校和正校长之间存在一对一的联系。

(2) 一对多联系 (1:n)。实体集 A 中的每一个实体与实体集 B 中的多个实体对应, 反之亦然, 记为 1:n。例如, 班级和学生两个实体型, 一个班级可以有多个学生, 而一名学生只属于一个班级。

一对多联系是最普遍的联系。也可以把一对一的联系看作一对多联系的一个特殊情况。

(3) 多对多联系 (m:n)。实体集 A 中的每一个实体与实体集 B 中的多个实体对应, 反之亦然, 记为 m:n。例如, 学生和课程两个实体型, 一个学生可以选修多门课程, 一个课程由多个学生选修。因此, 学生和课程间存在多对多的联系。

3.E-R 方法

E-R 图又被称为实体-联系图, 它提供了表示实体、属性和联系的方法, 用来描述现实世界的概念模型。构成 E-R 图的基本要素是实体、属性和联系, 其表示方法为:

(1) 实体: 用矩形表示, 矩形框内写明实体名。

(2) 属性: 用椭圆表示, 椭圆框内写明联系的名称并用无向边将其与相应的实体连接起来。

(3) 联系: 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)。

如图 1-4 所示, 为用 E-R 图表示教学实体模型。

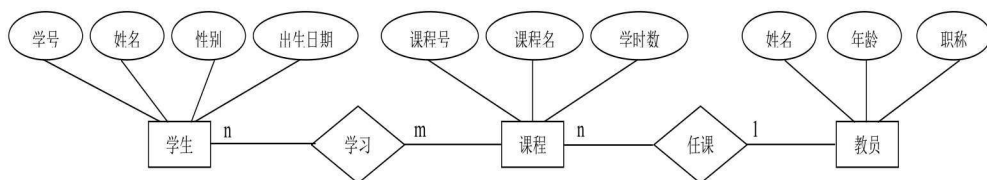


图 1-4 教学实体 E-R 图

1.3.2 逻辑数据模型

逻辑数据模型, 主要包括层次模型、网状模型、关系模型等, 它是按计算机系统的特点对数据建模, 主要用于 DBMS 的实现。

1.层次模型

层次模型用树状结构来表示各类实体及实体间的联系。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父结点, 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, 每个结点表示一个实体集, 实体集之间的联系用结点之间的连线 (有向边) 表示, 这种联系是父子之间的一对多联系。同一父结点的子结点称为兄弟结点, 没有子结点的结点称为叶结点。若需要子结点有很多父结点或不同父结点的子结点间联系, 则无法使用层次模式, 必须改用其他模式。

层次模型结构的优点是结构简单, 易于操作; 从上而下寻找数据容易; 与日常生活的数据类型相似。其缺点是寻找非直系的结点非常麻烦, 必须通过多个父结点由下而上, 再向下寻找, 搜寻的效率太低。

如图 1-5 所示, 给出了一个层次模型。图中, “系”为根结点; “教研室”和“课程”为兄弟结点, 是“系”的子结点; “教师”是“教研室”的子结点; “教师”和“课程”为叶结点。

2. 网状模型

用网状结构表示实体及其之间联系的模型称为网状模型。网状模型是层次模型的扩张, 去掉了层次模型的两个限制, 即允许结点有多于一个的父结点, 可以有一个以上的结点没有父结点。因此, 网状模型可以方便地表示各种类型的联系。

如图 1-6 所示, 给出了一个简单的网状模型。每一个联系都代表实体之间的一对多联系, 系统用单向或双向环形链接指针来具体实现这种联系。

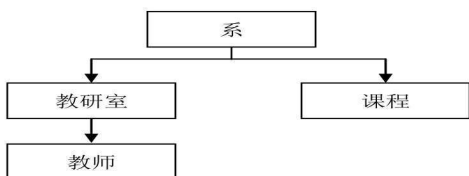


图 1-5 层次模型示例

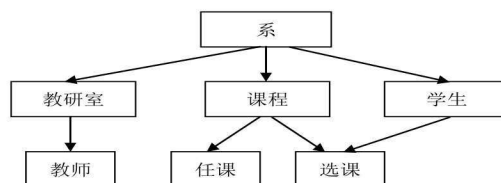


图 1-6 网状模型示例

网状模型的优点是表示多对多联系具有很大的灵活性, 这种灵活性是以数据结构复杂化为代价的。其缺点是路径太多, 当增加或删除数据时, 牵动的相关数据很多, 重建和维护数据比较麻烦。

网状模型与层次模型在本质上是一样的。从逻辑上看, 它们都是用结点表示实体, 用有向边表示实体间的联系, 实体和联系用不同的方法来表示; 从物理上看, 每一个结点都是一个存储记录, 用链接指针来实现记录之间的联系。这种用指针将所有数据记录都“捆绑”在一起的特点, 使得层次模型和网状模型存在难以实现系统修改与扩充等缺陷。

3. 关系模型

20 世纪 70 年代初提出了关系模型。关系模型是用二维表结构来表示实体以及实体之间联系的模型。在关系模型中, 操作的对象和结果都是二维表, 这种二维表就是关系。

关系模型与层次模型、网状模型的本质区别在于数据描述的一致性, 模型概念单一。在关系数据库中, 每一个关系都是一个二维表, 无论实体本身还是实体间的联系都用称为“关系”的二维表来表示, 使得描述实体的数据本身能够自然地反映它们之间的联系。而传统的层次和网状模型是使用链接指针来存储和体现联系的。支持关系模型的数据库管理系统称为关系数据库管理系统。从本质上说, Access 仍然是传统的关系数据库系统, 但它在用户界面、程序设计等方面进行了很好的扩充, 提供了面向对象程序设计的强大功能。

1.3.3 物理数据模型

物理数据模型 (简称物理模型) 是一种面向计算机物理表示的模型。物理数据模型给出了逻辑数据模型在计算机上物理结构的表示, 它是描述数据在储存介质上组织结构的数据模型, 它不但与具体的 DBMS 有关, 而且还与操作系统和硬件有关。每一种逻辑数据模型在实现时都有与其相对应的物理数据模型。

DBMS 为了保证其独立性与可移植性, 大部分物理数据模型的实现工作是由系统自动完成的, 而设计者只需设计索引等特殊结构。