

全国高等院校计算机基础教育“十三五”规划教材

MySQL

数据库程序设计

MySQL Shujuku Chengxu Sheji

◆ 何元清 魏 哲 主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

全国高等院校计算机基础教育“十三五”规划教材

MySQL 数据库程序设计

何元清 魏 哲 主 编

张 欢 张娅岚 周 敏 副主编

常州大学图书馆
藏书章

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

MySQL 是最受欢迎的开源关系数据库管理系统之一, 由于其性能优越、功能强大, 已经广泛应用于互联网上各类中小型网站及信息管理系统的应用开发, 受到广大软件爱好者及商业软件用户的青睐。

本书以讲解 MySQL 数据库基础知识为目标, 以完成学生信息管理案例的实现为载体, 深入讲解数据库基础知识、MySQL 编程、数据库和表的操作、视图管理、数据管理、PHP 的 MySQL 编程等内容。

本书内容丰富、讲解细致, 适合作为高等院校非计算机专业数据库程序设计的教材, 也可作为培训机构的培训教材和全国计算机等级考试(二级) MySQL 数据库程序设计的培训教材, 同时也是一本面向广大 MySQL 爱好者的实用参考书。

图书在版编目(CIP)数据

MySQL 数据库程序设计 / 何元清, 魏哲主编. —北京:
中国铁道出版社, 2018. 1 (2018. 7 重印)
全国高等院校计算机基础教育“十三五”规划教材
ISBN 978-7-113-24180-3

I. ①M… II. ①何… ②魏… III. ①SQL 语言-程序设计-高等学校-教材 IV. ①TP311. 132. 3

中国版本图书馆 CIP 数据核字 (2017) 第 330991 号

书 名: MySQL 数据库程序设计
作 者: 何元清 魏 哲 主编

策 划: 周海燕
责任编辑: 周海燕 彭立辉
封面设计: 乔 楚
责任校对: 张玉华
责任印制: 郭向伟

读者热线: (010) 63550836

出版发行: 中国铁道出版社 (100054, 北京市西城区右安门西街 8 号)

网 址: <http://www.tdpress.com/51eds/>

印 刷: 三河市航远印刷有限公司

版 次: 2018 年 1 月第 1 版 2018 年 7 月第 2 次印刷

开 本: 787mm×1092mm 1/16 印张: 15 字数: 362 千

书 号: ISBN 978-7-113-24180-3

定 价: 38.00 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社教材图书营销部联系调换。电话: (010) 63550836

打击盗版举报电话: (010) 51873659

随着科技的发展,计算机技术应用已经涉及人们生活的方方面面,对人们的生活方式产生了重要的影响。数据库技术是计算机技术的核心技术,支撑着整个计算机信息系统和应用系统,特别是随着大数据时代的到来,数据库技术已经成为当前计算机技术领域最活跃的版块之一。数据处理与应用能力已经成为大学生的基本素质之一,也关系到学生的择业及就业后对工作的适应能力。MySQL 是最受欢迎的开源关系数据库管理系统之一,由于其性能优越、功能强大,已经广泛应用于互联网上各类中小型网站及信息管理系统的应用开发,受到广大软件爱好者及商业软件用户的青睐。“MySQL 数据库程序设计”是根据教育部计算机基础教学指导委员会“1+X”培养要求开设的公共基础课程,该课程对学生的知识结构、素质的培养、智力的开发等变得越来越重要。对此,我们在多年教学实践的基础上,根据人才培养的新要求以及新时代教育技术和教学手段在教学改革中的应用现状和水平,编写了本书。

全书共分三篇 11 章,基础篇包括第 1~3 章,实践篇包括第 4~9 章,应用篇包括第 10~11 章,全面介绍了“MySQL 数据库程序设计”课程要求的各方面知识,包括数据库基础知识、MySQL 编程、数据库和表的操作、视图管理、数据管理以及 PHP 的 MySQL 编程等内容。全书以完成学生信息管理案例的实现为载体,内容系统、新颖、简明、实用。为帮助读者更好地掌握知识点和操作技能,每章最后配有习题,还同步编写了配套上机指导教材。

本书由何元清、魏哲任主编,张欢、张娅岚、周敏任副主编,傅强和刘晓东主审。其中,第 1~3 章由何元清编写,第 4~6 章由张娅岚编写,第 7、8 章由张欢编写,第 9 章由周敏、魏哲编写,第 10、11 章由魏哲编写。全书由何元清统稿、定稿。

本书在编写过程中得到中国民航飞行学院各级领导和同行专家的大力支持和帮助,计算机工程教研室罗银辉、刘光志、戴蓉、路晶、宋海军、华漫、徐国标在资料的收集和整理方面付出了辛勤的劳动。在编写过程中,中国民航飞行学院教务处也给予了大力支持,在此一并表示衷心感谢。

由于时间仓促,编者水平有限,书中难免存在疏漏和不妥之处,敬请读者批评指正。

编者

2017 年 11 月

基 础 篇

第 1 章 数据库系统概述	1
1.1 数据库基础	1
1.1.1 数据、信息、数据处理	1
1.1.2 数据库技术的产生与发展	2
1.2 数据库系统结构	4
1.2.1 数据库系统的基本概念	4
1.2.2 数据库系统的体系结构	5
1.3 数据模型	6
1.3.1 概述	7
1.3.2 概念模型	7
1.3.3 逻辑模型	8
1.3.4 关系数据库系统	10
习题	12
第 2 章 MySQL 编程基础	16
2.1 MySQL 概述	16
2.2 WampServer	18
2.2.1 WampServer 组件	18
2.2.2 WampServer 安装与配置	19
2.3 结构化查询语言 (SQL)	26
2.4 MySQL 数据基础	28
2.4.1 数据类型	28
2.4.2 常量	31
2.4.3 变量	32
2.4.4 运算符与表达式	34
2.4.5 函数	36
习题	40
第 3 章 程序设计基础	44
3.1 程序设计概述	44
3.2 PHP 编程基础	45
3.2.1 PHP 配置	46
3.2.2 PHP 基本语法	48
3.2.3 PHP 数据类型	49
3.2.4 PHP 表达式	49
3.3 PHP 流程控制	51

3.3.1 顺序结构	51
3.3.2 选择结构	51
3.3.3 循环结构	53
3.3.4 PHP 文件包含	56
习题	58

实 践 篇

第 4 章 创建数据库与数据表	62
4.1 数据库	62
4.1.1 创建并选择数据库	62
4.1.2 修改及查看数据库	64
4.1.3 删除数据库	65
4.2 数据表	65
4.2.1 创建数据表	65
4.2.2 修改数据表	68
4.2.3 查看表	72
4.2.4 复制表结构	73
4.2.5 删除表	74
4.3 数据完整性约束	74
习题	82
第 5 章 数据操纵	85
5.1 插入数据	85
5.2 删除数据	88
5.3 修改数据	89
习题	92
第 6 章 索引	96
6.1 索引简介	96
6.1.1 索引的概念	96
6.1.2 索引的利弊	96
6.2 索引的分类	97
6.3 创建、查看和删除索引	98
6.3.1 创建索引	98
6.3.2 查看索引	101
6.3.3 删除索引	102
习题	102
第 7 章 数据查询	105
7.1 SELECT 语句	105
7.2 列选择子句	106
7.3 FROM 子句与多表连接	113
7.4 WHERE 子句	118

7.5	GROUP BY 子句	130
7.6	HAVING 子句	131
7.7	ORDER BY 子句	132
7.8	LIMIT 子句	134
7.9	UNION 语句	134
	习题	135
第 8 章	视图	139
8.1	视图概述	139
8.2	视图的创建	140
8.3	视图结构的查看与修改	142
8.3.1	查看视图结构	142
8.3.2	修改视图结构	143
8.4	视图数据的查询与更新	144
8.4.1	视图数据的查询	144
8.4.2	视图数据的更新	145
8.5	视图的删除	148
	习题	148
第 9 章	数据管理	151
9.1	触发器	151
9.1.1	触发器介绍	151
9.1.2	创建触发器	151
9.1.3	查看触发器	154
9.1.4	删除触发器	157
9.1.5	使用触发器的注意事项	157
9.2	事件	158
9.2.1	事件介绍	158
9.2.2	如何开启事件调度器	158
9.2.3	创建事件	160
9.2.4	查看事件	162
9.2.5	修改事件	163
9.2.6	删除事件	165
9.3	存储过程和存储函数	165
9.3.1	创建存储过程和存储函数	166
9.3.2	查看存储过程和存储函数的定义	172
9.3.3	删除存储过程和存储函数	174
9.3.4	修改存储过程和存储函数	175
9.3.5	自定义错误处理程序	176
9.3.6	游标	179
9.4	访问控制	183
9.4.1	登录和退出 MySQL 服务器	183

9.4.2	创建用户账户	184
9.4.3	删除普通用户	185
9.4.4	修改普通用户账号	186
9.4.5	修改普通用户密码	187
9.4.6	账号权限管理	187
9.5	备份与恢复	192
9.5.1	使用 mysqldump 命令备份	192
9.5.2	使用 mysql 命令恢复	193
9.5.3	使用 SELECT INTO OUTFILE 导出文本文件	193
9.5.4	使用 LOAD DATA INFILE 导入文本文件	194
9.5.5	使用图形界面备份和恢复数据	194
9.5.6	使用二进制日志还原数据库	196
习题		198

应 用 篇

第 10 章	PHP 的 MySQL 编程	203
10.1	编程步骤	203
10.2	连接 MySQL 数据库服务器	203
10.3	选择 MySQL 数据库	204
10.4	操作 MySQL 数据库	205
10.4.1	数据的添加	206
10.4.2	数据的查询	206
10.4.3	数据的修改	210
10.4.4	数据的删除	212
10.5	关闭 MySQL 服务器的连接	213
10.6	实例	214
10.6.1	动态添加用户信息	214
10.6.2	动态删除用户信息	216
习题		217
第 11 章	开发实例	218
11.1	需求分析	218
11.2	系统设计	218
11.3	数据库设计	219
11.4	系统实现	220
11.4.1	学生基本信息管理系统页面实现	221
11.4.2	添加学生信息页面实现	221
11.4.3	查看学生信息页面实现	223
11.4.4	删除学生信息页面实现	224
11.5	学生信息管理系统程序清单	226
习题		232

第1章 数据库系统概述

本章导读

数据库技术从诞生到现在,在不到半个世纪的时间里,形成了坚实的理论基础、成熟的商业产品和广泛的应用领域,吸引了越来越多的研究者加入,使得数据库成为一个研究者众多且被广泛关注的研究领域。随着信息管理内容的不断扩展和新技术的层出不穷,数据库技术面临着前所未有的挑战。那么什么是数据库?如何管理人们面临的越来越多的信息内容?本章将具体进行介绍。

学习目标

- 了解数据库技术的发展、特点。
- 掌握数据库的基本知识、体系结构。
- 理解数据库模型。

1.1 数据库基础

1.1.1 数据、信息、数据处理

数据、信息和数据处理是与数据库密切相关的3个基本概念。

1. 数据

人们通常使用各种各样的物理符号来表示客观事物的特征,这些符号及其组合就是数据。数据的概念包括两方面:数据内容和数据形式。数据内容是指所描述客观事物的具体特性,也就是通常所说的数据的“值”;数据形式则是指数据内容存储在媒体上的具体形式,也就是通常所说的数据的“类型”。数据主要有数字、文字、声音、图形和图像等多种形式。

2. 信息

信息是指数据经过加工处理后获取的有用知识。信息是客观事物属性的反映,是有用的数据。信息无处不在,它存在于人类社会的各个领域,而且不断变化,人们需要不断获取信息、加工信息,运用信息为社会的各个领域服务。

数据和信息是两个相互联系但又相互区别的概念。数据是信息的具体表现形式;信息是数据有意义的表现,是数据的内涵,是对数据语义的解释。

3. 数据处理

数据处理也称信息处理,就是将数据转换为信息的过程。数据处理的内容主要包括:数据的收集、整理、存储、加工、分类、维护、排序、检索和传输等。数据处理的目的是从大量的数据

中,根据数据自身的规律及其相互联系,通过分析、归纳、推理等科学方法,利用计算机技术、数据库技术等技术手段,提取有效的信息资源,为进一步分析、管理和决策提供依据。

例如,学生各门课程成绩为原始数据,经过计算得出平均成绩和总成绩等信息,这个计算处理的过程就是数据处理。

1.1.2 数据库技术的产生与发展

计算机数据处理技术与其他技术的发展一样,经历了由低级到高级的发展过程。计算机数据管理随着计算机硬件(主要是外存储器)、软件技术和计算机应用范围的发展而不断发展,管理水平不断提高,管理方式也发生了很大的变化。数据库管理技术的发展主要经历了人工管理、文件管理和数据库管理3个阶段。

1. 人工管理阶段

早期的计算机主要用于科学计算,计算处理的数据量很小,基本上不存在数据管理的问题。20世纪50年代初,计算机开始应用于数据处理。当时的计算机没有专门管理数据的软件,也没有像磁盘这样可随机存取的外部存储设备,对数据的管理也没有一定的格式。数据依附于处理它的应用程序,使数据和应用程序一一对应,互为依赖。

由于数据与应用程序的对应、依赖关系,某应用程序中的数据无法被其他程序利用,程序与程序之间存在着大量重复数据,即数据冗余;同时,由于数据是对应某一应用程序的,使得数据的独立性很差,如果数据的类型、结构、存取方式或输入/输出方式发生变化,处理它的程序必须相应改变,数据结构性差,而且数据不能长期保存。

在人工管理阶段存在的主要问题如下:

- ① 数据不具有独立性,程序和数据一一对应。
- ② 数据不保存,包含在程序中。数据在程序运行完后和程序一起释放。
- ③ 数据需要程序自己管理,没有进行数据管理的软件。
- ④ 数据不共享,一组数据只能对应一个程序。

人工管理阶段程序与数据的对应关系如图1-1所示。



图 1-1 人工管理阶段程序与数据的对应关系

2. 文件管理阶段

从20世纪50年代后期开始至60年代末为文件管理阶段。应用程序通过操作系统的文件管理功能来管理数据。由于计算机存储技术的发展和软件系统的进步,如计算机硬件出现了可直接存取的磁盘、磁带及磁鼓等外部存储设备;软件出现了高级语言和操作系统,数据处理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，形成数据的文件管理方式。

在文件管理阶段，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式来存取、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。但文件系统只是简单地存储数据，数据的存取在很大程度上仍依赖于应用程序，不同程序难于共享同一数据文件。与早期的人工管理阶段相比，利用文件系统管理数据的效率和数量都有很大的提高，但仍存在以下问题：

① 数据独立性较差，没有完全独立。

② 存在数据冗余。

③ 数据不能集中管理。

文件管理阶段应用程序与数据之间的关系如图 1-2 所示。

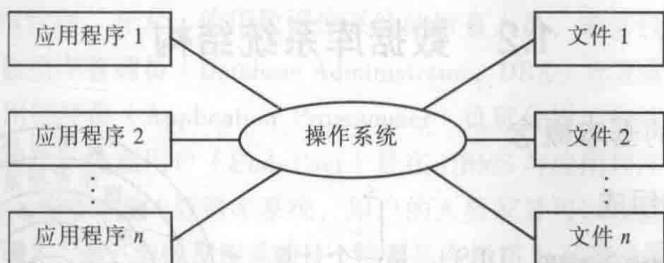


图 1-2 文件管理阶段应用程序与数据之间的关系

3. 数据库管理阶段

数据库管理阶段是 20 世纪 60 年代末在文件管理基础上发展起来的。随着计算机系统性价比的持续提高、软件技术的不断发展，人们克服了文件系统的不足，开发了一类新的数据管理软件——数据库管理系统（Database Management System, DBMS），运用数据库技术进行数据管理，将数据管理技术推向了数据库管理阶段。

数据库技术使数据有了统一的结构，对所有的数据实行统一、集中、独立的管理，以实现数据的共享，保证数据的完整性和安全性，提高了数据管理效率。数据库也是以文件方式存储数据的，但它是数据的一种高级组织形式。在应用程序和数据库之间，由 DBMS 把所有应用程序中使用的相关数据汇集起来，按统一的数据模型，以记录为单位存储在数据库中，为各个应用程序提供方便、快捷的查询、操纵。

数据库系统与文件系统的区别：数据库中数据的存储是按同一结构进行的，不同的应用程序都可直接操作使用这些数据，应用程序与数据间保持高度的独立性；数据库系统提供了一套有效的管理手段，保持数据的完整性、一致性和安全性，使数据具有充分的共享性；数据库系统还为用户管理、控制数据的操作，提供了功能强大的操作命令，用户可通过直接使用命令或将命令嵌入应用程序中，简单方便地实现数据的管理、控制操作。

数据库管理阶段的主要特点：

① 实现了数据结构化。

② 实现了数据共享。

③ 实现了数据独立。

④ 实现了数据的统一控制。

数据库管理阶段应用程序与数据之间的关系如图 1-3 所示。

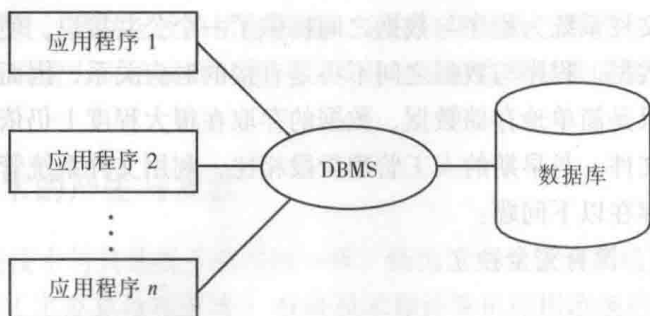


图 1-3 数据库管理阶段应用程序与数据之间的关系

1.2 数据库系统结构

1.2.1 数据库系统的基本概念

1. 数据库系统的组成

数据库系统 (Database System, DBS), 是一个计算机应用系统。它由数据库、数据库管理系统、计算机硬件、计算机软件 (包括操作系统、语言及编译系统等) 和用户等部分组成, 如 1-4 所示。

(1) 数据库

数据库 (Database, DB) 是指数据库系统中按一定的组织形式存储在一起的相互关联的数据的集合。数据库中的数据也是以文件的形式存储在存储介质上的, 它是数据库系统操作的对象和结果。数据库中的数据具有集中性和共享性。所谓集中性是指把数据库看成性质不同的数据文件的集合, 其中的数据冗余很小。所谓共享性是指多个不同用户使用不同语言, 为了不同应用目的可同时存取数据库中的数据。

数据库中的数据由 DBMS 进行统一管理和控制, 用户对数据库进行的各种数据操作都是通过 DBMS 实现的。

(2) 数据库管理系统

数据库管理系统 (DBMS) 是指负责数据库存取、维护、管理的系统软件。DBMS 提供对数据库中数据资源进行统一管理和控制的功能, 将用户应用程序与数据库数据相互隔离。它是数据库系统的核心, 其功能的强弱是衡量数据库系统性能优劣的主要指标。它主要包括如下功能:

- ① 数据库定义 (描述) 功能。
- ② 数据库操纵功能。
- ③ 数据库管理功能。
- ④ 通信功能。

(3) 计算机硬件

计算机硬件 (Hardware) 是数据库系统赖以存在的物质基础, 是存储数据及运行数据库管理

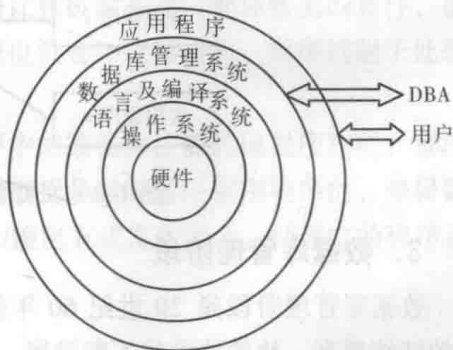


图 1-4 数据库系统的组成

系统的硬件资源,主要包括主机、存储设备、I/O 通道等。大型数据库系统一般都建立在计算机网络环境下。为使数据库系统获得较满意的运行效果,应对计算机的 CPU、内存、磁盘、I/O 通道等技术性能指标进行较高的配置。

(4) 计算机软件

软件系统包括支持数据库管理系统运行的操作系统(如 Windows XP/7/10 等)、开发应用程序的高级语言及其编译系统等。

(5) 应用程序

应用程序(Application)是在 DBMS 的基础上,由用户根据应用的实际需要所开发的、处理特定业务的程序。应用程序的操作范围通常仅是数据库的一个子集,即用户所需的那部分数据。

(6) 用户

用户(User)是指管理、开发、使用数据库系统的所有人员,通常包括数据库管理员、应用程序员和终端用户。数据库管理员(Database Administrator, DBA)负责管理、监督、维护数据库系统的正常运行;应用程序员(Application Programmer)负责分析、设计、开发、维护数据库系统中运行的各类应用程序;终端用户(End-User)是在 DBMS 与应用程序支持下,操作使用数据库系统的普通使用者。不同规模的数据库系统,用户的人员配置可以根据实际情况有所不同,大多数用户都属于终端用户。在小型数据库系统中,特别是在微机上运行的数据库系统中,通常 DBA 由终端用户担任。

2. 数据库系统的特点

数据库系统的出现是计算机数据处理技术的重大进步,它具有以下特点:

① 数据共享:指多个用户可以同时存取数据而不相互影响。数据共享包括以下三方面:所有用户可以同时存取数据;数据库不仅可以为当前的用户服务,也可以为将来的新用户服务;可以使用多种语言完成与数据库的接口。

② 减少数据冗余:数据冗余就是数据重复,既浪费存储空间,又容易导致数据不一致。在非数据库系统中,由于每个应用程序都有自己的数据文件,所以数据存在着大量的重复。

③ 具有较高的数据独立性:所谓数据独立是指数据与应用程序之间彼此独立,不存在相互依赖的关系。应用程序不必随数据存储结构的改变而变动,这是数据库一个最基本的优点。

④ 增强了数据安全性和完整性:数据库加入的安全保密机制,可以防止对数据的非法存取。数据库的集中控制方式,有利于控制数据的完整性;数据库系统采取的并发访问控制,保证了数据的正确性;另外,数据库系统还采取了一系列措施,实现了对数据库破坏的恢复。

3. 数据库应用系统

数据库应用系统(Database Application System, DBAS)是在 DBMS 支持下根据实际问题开发出来的数据库应用软件,它包括数据库和应用程序两部分,需要在 DBMS 支持下开发。由于数据库的数据要被不同的应用程序共享,因此在开发应用程序之前要先设计数据库,然后开发应用程序。应用程序的开发可通过“功能分析—总体设计—详细设计—编码—调试”等步骤来实现。

1.2.2 数据库系统的体系结构

为了有效地组织、管理数据,提高数据库的逻辑独立性和物理独立性,人们为数据库设计了一个严谨的体系结构,包括 3 个模式(模式、外模式和内模式)和 2 个映射(外模式-模式映射

和模式-内模式映射)。美国 ANSI/X3/SPARC 的数据库管理系统研究小组于 1975 年、1978 年提出了标准化的建议,将数据库结构分为 3 级:面向用户或应用程序员的用户级;面向建立和维护数据库人员的概念级;面向系统程序员的物理级。用户级对应外模式,概念级对应模式,物理级对应内模式,所以不同级别的用户对数据库可以形成不同的视图。

1. 模式

模式又称概念模式或逻辑模式,对应于概念级。它是由数据库设计者综合所有用户的数据,按照统一的观点构造的全局逻辑结构,是对数据库中全部数据的逻辑结构和特征的总体描述,是所有用户的公共数据视图(全局视图)。它是由数据库系统提供的模式数据描述语言(Data Description Language, DDL)来描述、定义的,体现、反映了数据库系统的整体观。

2. 外模式

外模式又称子模式,对应于用户级。它是某个或某几个用户所看到的数据库的数据视图,是与某一应用有关的数据的逻辑表示。外模式是从模式导出的一个子集,包含模式中允许特定用户使用的那部分数据。用户可以通过外模式描述语言(外模式 DDL)来描述、定义用户的数据记录,也可以利用数据操纵语言(Data Manipulation Language, DML)对这些数据记录进行操作。外模式反映了数据库的用户观。

3. 内模式

内模式又称存储模式,对应于物理级。它是数据库中全体数据的内部表示或底层描述,是数据库最低一级的逻辑描述,它描述了数据在存储介质上的存储方式和物理结构,对应着实际存储在外存储介质上的数据库。内模式由内模式描述语言(内模式 DDL)来描述、定义,它是数据库的存储观。

4. 三级模式间的映射

数据库系统的三级模式是数据在 3 个级别(层次)上的抽象,使用户能够逻辑地、抽象地处理数据而不必关心数据在计算机中的物理表示和存储。实际上,对于一个数据库系统而言,只有物理级数据库是客观存在的,它是进行数据库操作的基础;概念级数据库不过是物理数据库的一种逻辑的、抽象的描述(即模式);用户级数据库则是用户与数据库的接口,它是概念级数据库的一个子集(外模式)。

用户应用程序根据外模式进行数据操作,通过外模式-模式映射,定义和建立某个外模式与模式间的对应关系,将外模式与模式联系起来,当模式发生改变时,只要改变其映射,就可以使外模式保持不变,对应的应用程序也可保持不变;另一方面,通过模式-内模式映射,定义建立数据的逻辑结构(模式)与存储结构(内模式)间的对应关系,当数据的存储结构发生变化时,只需改变模式-内模式映射,就能保持模式不变,因此应用程序也可以保持不变。

1.3 数据模型

数据模型是现实世界数据特征的抽象,用于描述一组数据的概念和定义,数据模型按应用层次又分为概念模型和逻辑模型。概念模型是面向客观世界和用户的模型,用于数据库设计;逻辑模型是面向计算机系统的模型,用于数据库管理系统的实现。

1.3.1 概述

为方便计算机处理现实世界中的具体事物，必须把具体事物转换成计算机能够处理的数据。在数据库系统中，基本思想是把现实世界中的客观事物抽象为概念数据模型，然后再将概念数据模型转换成某一数据库管理系统支持的逻辑数据模型，如图 1-5 所示。

概念模型是数据库设计人员进行数据库设计的一种重要工具，也是数据库设计人员和用户之间进行交流的语言，E-R 模型是最常使用的一种概念模型。逻辑模型是面向数据库系统的模型，用于 DBMS 的实现，数据库管理系统常用的逻辑模型包括层次模型、网状模型和关系模型。

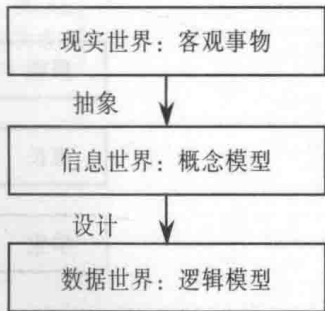


图 1-5 客观事物到数据的抽象过程

1.3.2 概念模型

概念数据模型与 DBMS 无关的，主要用于信息世界的建模，是现实世界到信息世界的第一层抽象。概念模型应该具有较强的语义表达能力，能够方便、直接地表达应用中的各种语义知识，同时也具有简单、清晰、易于用户理解等特点。

1. 基本概念

(1) 实体

实体是存在于现实世界中并且可以与其他物体区分开的物体，它可以是一个具体事物，如一个人、一辆汽车等，也可以是抽象的事物，如一个表、一个项目等。

(2) 属性

属性用于描述实体的特征，一个实体可由若干个属性来表述。例如，办公桌有长、宽、高、颜色、重量等属性；学生有学号、姓名、性别、出生日期等属性。

(3) 联系

在现实世界中，事物之间都存在相互联系。联系是实体集之间关系的抽象表示，实体集之间的联系分为 3 种：

① 一对一联系 (1:1)：对于两个实体集 A 和 B ，若 A 中的每一个值在 B 中至多有一个实体值与之对应，反之亦然，则称实体集 A 和 B 具有一对一的联系。例如，一个班级只有一个班长，而一个班长只在一个班级，则班级与班长之间具有一对一的联系。

② 一对多联系 (1:N)：对于两个实体集 A 和 B ，若 A 中的每一个值在 B 中有多个实体值与之对应，反之 B 中每一个实体值在 A 中至多有一个实体值与之对应，则称实体集 A 和 B 具有一对多的联系。例如，一个班长可以有多个同学，但是每个学生只能有一个班长，则班长与学生之间存在一对多的联系。

③ 多对多联系 (M:N)：对于两个实体集 A 和 B ，若 A 中每一个实体值在 B 中有多个实体值与之对应，反之亦然，则称实体集 A 与实体集 B 具有多对多的联系。例如，一个学生可以选修多门课程，而每门课程可以有多个学生选修，则学生与课程之间存在多对多的联系。

2. 表示方法

通常用实体-联系方法表示概念模型，该方法用 E-R 模型来描述，如图 1-6 所示。

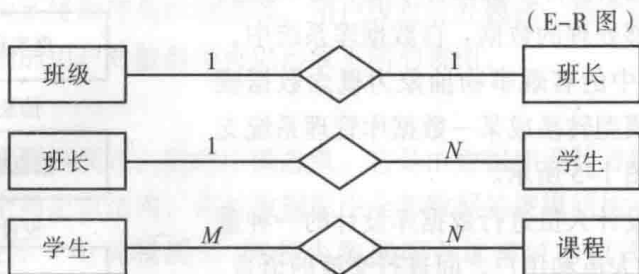


图 1-6 E-R 模型

E-R 图提供了表示信息世界中实体、属性和联系的方法。

- ① 实体用矩形框表示，框中描述实体名。
- ② 联系用菱形框表示，框中描述联系名。
- ③ 属性用椭圆形框表示，框中描述属性名。
- ④ 连线用于描述实体与属性、实体与联系、联系与属性之间的连接。

1.3.3 逻辑模型

逻辑模型是用户从数据库所看到的模型，是具体的 DBMS 所支持的数据模型，分为 3 种：层次模型、网状模型、关系模型。

1. 层次模型

用树形结构表示数据及其联系的数据模型称为层次模型 (Hierarchical Model)。树是由结点和连线组成，结点表示数据集，连线表示数据之间的联系，树形结构只能表示一对多联系。通常将表示“一”的数据放在上方，称为父结点；而表示“多”的数据放在下方，称为子结点。树的最高位置只有一个结点，称为根结点。根结点以外的其他结点都有且仅有一个父结点与它相连，同时可能有一个或多个子结点与它相连。没有子结点的结点称为叶结点，它处于分支的末端。

层次模型的基本特点：

- ① 有且仅有一个结点无父结点，称其为根结点。
- ② 其他结点有且只有一个父结点。

支持层次数据模型的 DBMS 称为层次数据库管理系统，在这种系统中建立的数据库是层次数据库。层次模型可以直接方便地表示一对一联系和一对多联系，但不能用它直接表示多对多联系。采用层次模型结构的数据库的典型代表是 IBM 公司的 IMS (Information Management System) 数据库管理系统。该系统于 1966 年推出，是第一个大型商用数据库管理系统，曾经得到广泛应用。例如，学院机构就是一个典型的层次数据模型，如图 1-7 所示。

2. 网状模型

用网络结构表示数据及其联系的数据模型称为网状模型 (Network Model)。网状模型是层次模型的拓展，网状模型的结点间可以任意发生联系，能够表示各种复杂的联系。

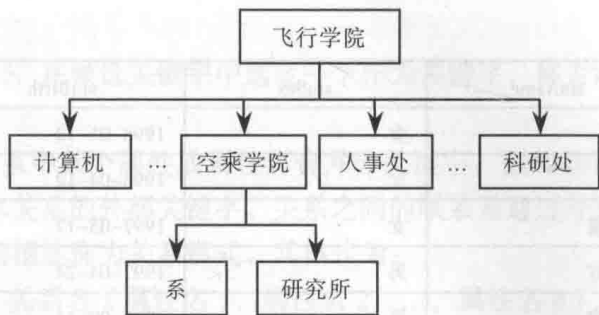


图 1-7 学院机构示意图

网状模型的基本特点：

- ① 一个以上结点无父结点。
- ② 至少一个结点有多于一个的父结点。

网状模型和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数据，用连线表示数据间的联系。从物理上看，层次模型和网状模型都是用指针来实现两个结点之间的联系。层次模型是网状模型的特殊形式，网状模型是层次模型的一般形式。

支持网状模型的 DBMS 称为网状数据库管理系统,在这种系统中建立的数据库是网状数据库。网状结构可以直接表示多对多联系，这也是网状模型的主要优点。采用网状模型结构的数据库的典型代表是 20 世纪 70 年代数据系统语言研究会（Conference on Data System Language, CODASYL）下属的数据库任务组（Database Task Group, DBTG）提出的一个系统方案，即 DBTG 系统。例如，学生和课程的关系，就是一个典型的网状模型，如图 1-8 所示。

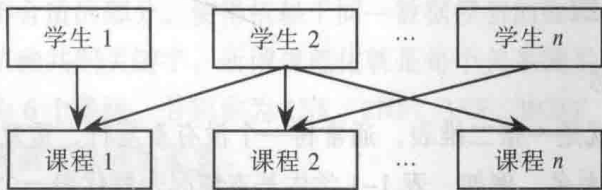


图 1-8 学生和课程的网状模型图

3. 关系模型

人们习惯用表格形式表示一组相关的数据，既简单又直观，表 1-1 所示为一张学生基本情况表（stu）。这种由行与列构成的二维表，在数据库理论中称为关系，用关系表示的数据模型称为关系模型（Relational Model）。在关系模型中，实体和实体间的联系都是用关系表示的，也就是说，二维表格中既存放着实体本身的数据，又存放着实体间的联系。关系不但可以表示实体间一对多的联系，通过建立关系间的关联，也可以表示多对多的联系。

表 1-1 学生基本情况表

stuID	stuName	stuSex	stuBirth	stuSchool
20160111001	王小强	男	1997-08-17	飞行技术学院
20160111002	何金品	男	1998-06-12	飞行技术学院
20160211011	李红梅	女	1997-07-19	交通运输学院
20160310022	张志斌	男	1998-07-10	航空工程学院