

主 审 赵立强

数据库原理与应用技术

—— MySQL

主 编 王 纲 曾晓宁 曹丽君



 河北科学技术出版社



P R E F A C E

前 言

数据库技术是涉及如何存储、管理和分析数据的方法和工具，它是计算机科学技术中的一个重要分支，也是应用范围最广、实用性最强的技术之一，它已成为信息社会的核心技术和重要基础。因此从人才培养、专业发展的角度出发，编写具有系统性、实用性的数据库教材是很有必要的。

本书内容由浅入深、循序渐进，系统地介绍了数据库技术的基础理论、基本技术和方法，具体包括数据库和数据表的操作、数据库完整性操作、数据的查询与更新、数据库编程、数据库设计等内容。全书以模块作为教学单元，每个单元又包含若干个任务，将数据库的各种技术渗透到每个单元的任务中，这样以任务驱动组织教学，通过完成具体的任务逐步引导读者掌握数据库应用的各种技术，为数据库应用系统的设计与开发打下坚实的基础。

本书中选用的数据库管理工具是 MySQL，它是最流行的关系数据库管理系统之一，具有体积小、速度快、总体成本低的特点，广泛应用于互联网应用中的数据存储，尤其是它具备开放源代码的优势，使它迅速成为中小型企业 and 网站应用的首选数据库。另外本书注重理论的可视化，对于实例的讲解都提供了相应的截屏图，每一个实例都给出运行结果。这样可读性更强，也便于读者自学。

本书可作为各类高等院校或高职院校计算机科学与技术、人工智能、大数据技术等相关专业数据库技术相关课程的教材，也可作为相关职业技能培训的教材，还可供从事数据库应用系统开发工作的人员学习参考。另外本书有配套的技能实训教材，该教材中提供了各教学单元对应的习题及答案、实验指导，还有五套综合练习题，为学生自主实验的顺利开展提供了良好的平台支撑，也有助于提升学生解决实际问题的能力。

参加本书的编写人员均为多年从事计算机教学的一线教师，具有丰富的教学经验，书中很多实例素材均来自于平时教学，具有很强的实用性。本书由王纲、曾晓宁、曹丽君任主编，负责全书的策划与统稿。张文娟、陈翠格、段永梅、韦澍芑为副主编，杨培肖、李佳瑞、董丽霞参编。另外本书在编写过程中，也参阅了大量专家学者的著作及互



联网上一些佚名作者的材料，受益匪浅，在此对他们表示深深的谢意。由于编者编写时间紧、任务重，书中难免存在错误与疏漏，敬请有关专家、学者和广大师生批评指正，以便不断修订完善。

编 者

2024年2月



CONTENTS

目 录

项目一 绪论	001
任务一 数据库系统概述	002
子任务一 数据库的基本概念	002
子任务二 数据管理技术的产生与发展	004
子任务三 数据库系统的特点	005
任务二 数据模型	006
子任务一 数据模型的组成要素	006
子任务二 概念模型	007
子任务三 逻辑模型	008
子任务四 物理模型	008
子任务五 层次模型	009
子任务六 网状模型	010
子任务七 关系模型	011
任务三 数据库系统的结构	012
子任务一 数据库系统模式	012
子任务二 数据库系统的三级模式	013
子任务三 数据库系统的二级映像功能与数据独立性	015
项目二 MySQL工具使用	018
任务一 MySQL概述	019
任务二 MySQL的安装与配置	023
任务三 MySQL服务的启动与停止	035
任务四 MySQL服务器的登录	039
子任务一 使用MySQL自带的客户端登录	039
子任务二 使用命令行方式登录	040
子任务三 数据库图形化管理工具 (Navicat Premium)	043



项目三 MySQL基础知识	049
任务一 结构化查询语言	050
子任务一 SQL概述	050
子任务二 SQL产生与发展	051
子任务三 SQL的特点	052
任务二 MySQL数据类型	054
子任务一 数据类型简介	054
子任务二 整数类型	055
子任务三 小数类型	057
子任务四 日期和时间类型	060
子任务五 字符串类型	063
任务三 MySQL运算符	068
子任务一 算术运算符	069
子任务二 比较运算符	072
子任务三 逻辑运算符	075
子任务四 位运算符	079
子任务五 运算符的优先级	080
任务四 MySQL函数	082
子任务一 聚合函数	082
子任务二 数学函数	084
子任务三 字符串函数	091
子任务四 日期和时间函数	099
子任务五 系统信息函数	105
子任务六 其他函数	107
项目四 MySQL数据库管理	110
任务一 创建数据库	111
子任务一 使用CREATE DATABASE语句创建数据库	111
子任务二 使用Navicat创建数据库	115

任务二 查看数据库	119
子任务一 使用SHOW查看数据库	119
子任务二 使用Navicat查看数据库	121
任务三 选择当前数据库	122
子任务一 使用USE选择数据库	123
子任务二 使用Navicat选择数据库	124
任务四 修改数据库	125
子任务一 使用ALTER修改数据库	126
子任务二 使用Navicat修改数据库	128
任务五 删除数据库	130
子任务一 使用DROP删除数据库	130
子任务二 使用Navicat删除数据库	131
项目五 MySQL数据表管理	135
任务一 创建数据表	136
子任务一 使用CREATE TABLE创建数据表	136
子任务二 使用Navicat创建数据表	139
任务二 查看数据表结构	141
子任务一 使用SHOW COLUMNS查看数据表结构	141
子任务二 使用Navicat查看数据表结构	142
任务三 修改数据表结构	143
子任务一 使用ALTER TABLE修改数据表结构	143
子任务二 使用Navicat修改数据表结构	144
任务四 重命名数据表	145
子任务一 使用RENAME TABLE重命名数据表	145
子任务二 使用Navicat重命名数据表	146
任务五 复制数据表	147
子任务一 使用CREATE TABLE ...LIKE复制数据表	147
子任务二 使用Navicat复制数据表	148



任务六 删除数据表	149
子任务一 使用 DROP TABLE 删除数据表	149
子任务二 使用 Navicat 删除数据表	149
项目六 MySQL 数据库完整性	152
任务一 实体完整性	153
子任务一 实体和实体完整性概念	153
子任务二 实体完整性的实现	154
任务二 参照完整性	160
子任务一 参照完整性	160
子任务二 参照完整性的实现	162
任务三 用户定义的完整性	164
子任务一 用户自定义的完整性	164
子任务二 默认值约束	166
任务四 完整性约束命名子句	168
子任务一 完整性约束命名子句	168
项目七 MySQL 数据查询与更新	171
任务一 单表查询	172
子任务一 简单查询	173
子任务二 条件查询	176
子任务三 使用 ORDER BY 查询排序	180
子任务四 使用 GROUP BY 子句分组查询	182
任务二 连接查询	184
任务三 嵌套查询	187
任务四 插入数据	189
任务五 修改数据	193
任务六 删除数据	194
项目八 MySQL 索引与视图	198
任务一 索引	199

子任务一 创建索引	199
子任务二 删除索引	204
任务二 视图	205
子任务一 认识视图	205
子任务二 创建视图	206
子任务三 查看视图	209
子任务四 使用视图	210
子任务五 修改视图	213
子任务六 删除视图	214
项目九 MySQL的数据库编程	217
任务一 触发器	218
子任务一 触发器概述	218
子任务二 创建触发器	219
子任务三 查看触发器	224
子任务四 删除触发器	225
任务二 事件	226
子任务一 事件概述	226
子任务二 创建事件	227
子任务三 查询事件	228
子任务四 启动与关闭事件	230
子任务五 修改事件	231
子任务六 删除事件	232
任务三 存储过程	233
子任务一 认识存储过程	233
子任务二 创建存储过程	234
子任务三 调用存储过程	240
子任务四 修改存储过程	242
子任务五 查看存储过程	244



子任务六 删除存储过程	245
任务四 存储函数	246
子任务一 认识存储函数	246
子任务二 创建存储函数	247
子任务三 调用存储函数	248
子任务四 查看存储函数	250
子任务五 修改存储函数	252
子任务六 删除存储函数	253
项目十 数据库设计	256
任务一 需求分析	257
子任务一 需求分析的内容	257
子任务二 需求分析的工具	258
任务二 概念结构设计	260
任务三 逻辑结构设计	261
子任务一 E-R图向关系模型转换的规则	261
子任务二 关系模型的优化及设计用户子模式	262
任务四 物理结构设计	263
子任务一 物理结构设计的内容	263
子任务二 关系模式存取方法选择	264
子任务三 确定数据库的存储结构	265
任务五 数据库的实施和维护	265
子任务一 数据库实施	265
子任务二 数据库运行维护	267
任务六 关系模式的规范化	268
子任务一 关系模式的范式	268
项目十一 数据库安全	272
任务一 数据库安全概述	273
任务二 用户管理	274

子任务一 创建用户	275
子任务二 修改用户	278
子任务三 删除用户	281
任务三 权限管理	283
子任务一 分配权限	283
子任务二 收回权限	285
任务四 角色管理	287
项目十二 MySQL数据库的备份和恢复	291
任务一 数据库备份概述	292
任务二 使用MYSQLDUMP命令进行备份	295
任务三 MySQL数据恢复	297

项目一

绪论



项目导读

数据库是数据管理的有效技术，是计算机科学的重要分支。要了解数据库的基本原理和应用，需要先从最基本的概念说起，在本项目中将介绍数据库的基本知识、数据管理技术的发展过程、数据库系统的组成部分以及一些数据模型等，为后面各项目的学习做好准备。



知识目标

1. 了解数据库概念和数据处理发展阶段。
2. 掌握数据库系统的特点与结构。
3. 理解概念模型、逻辑模型和物理模型等数据模型的概念。
4. 掌握层次模型、网状模型、关系模型等数据模型的概念。
5. 熟悉关系模型、表、列、关键字、候选键和外部关键字、域、数据类型等常用关系术语。



能力目标

1. 理解数据库概念，了解数据处理发展阶段。
2. 掌握数据库系统的特点与结构。
3. 理解概念模型、逻辑模型和物理模型等数据模型的概念。
4. 掌握层次模型、网状模型、关系模型等数据模型各自的特点。
5. 培养熟练运用关系模型、表、列、关键字、候选键和外部关键字、域、数据类型等常用关系术语的能力。



素质目标

1. 了解我国数据库技术的发展，树立强国意识。
2. 增强学习信心，提升自身的职业素养。

任务一 数据库系统概述

子任务一 数据库的基本概念

一、任务描述

我们总能在各种场合听到“信息”“数据”这些词，它们在计算机中具体指的是什么？要管理各种各样的数据，有什么高效的方法吗？为了解决这些问题，就要学习数据库技术。

二、预备知识

数据、数据库、数据库管理系统和数据库系统是与数据库技术密切相关的4个基本概念。我们就从基本概念出发来学习数据库的相关技术和应用。

三、任务实施

1. 数据 (data)

在现代计算机系统中数据的概念是广义的，早期的计算机系统主要用于科学计算，处理

的数据是数值型数据，现在计算机存储和处理的对象十分广泛，表示这些对象的数据也随之变得越来越复杂。

我们可以对数据做如下定义：数据是反映客观事物的状态、特征、特性的物理符号，能够被计算机存储和处理。它可以是数字，也可以是文字、图形、图像、音频、视频等，数据有多种表现形式，但它们都可以经过数字化后存入计算机。

2. 数据库 (DataBase, DB)

数据库，顾名思义，是存放数据的仓库。只不过这个仓库是在计算机存储设备上，而且数据是按一定的格式存放的。

严格地讲，数据库是指以一定的结构组织并长期存储在计算机存储器中，结构化、可共享且又独立于应用程序的相关数据的集合。概括地讲，数据库数据具有永久存储、有组织和可共享三个基本特点。

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

了解了数据和数据库的概念，下一个问题就是如何科学地组织和存储数据，如何高效地获取和维护数据，这就需要有一个系统软件——数据库管理系统。

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，数据库管理系统和操作系统一样是计算机的基础软件，也是一个大型复杂的软件系统，用于建立、使用和维护数据库，数据库管理系统是数据库系统的核心，是管理数据库的软件。

4. 数据库系统 (DataBase System, DBS)

数据库系统就是指计算机系统中引用了数据库技术后的完整系统，主要由计算机硬件（主机存储介质等）、软件（操作系统、数据库管理系统、开发工具等）、数据库和用户（数据库管理员、终端用户等）四个主要部分组成。

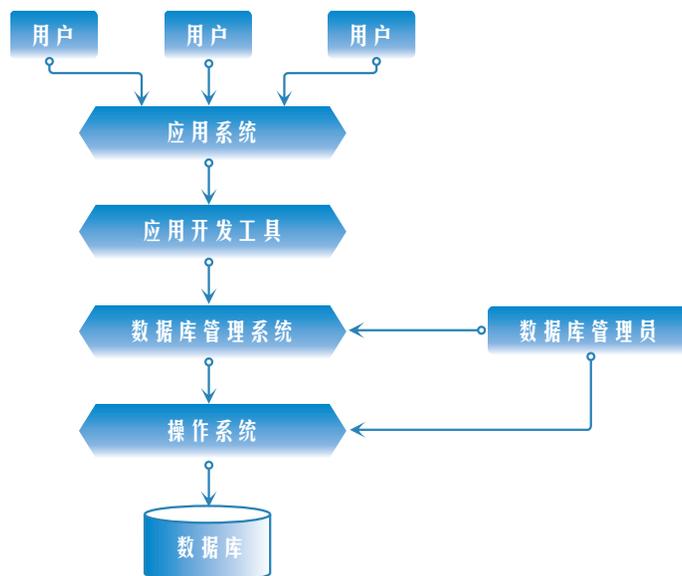


图 1-1 数据库系统的组成



子任务二 数据管理技术的产生与发展

一、任务描述

数据管理技术的出现和发展是伴随着计算机技术的发展而产生的，从计算机诞生以来，数据就是计算机计算和处理的对象，也是计算机信息存储的具体形式。计算机应用领域从早期的科学计算，到办公自动化的文档处理，再到多媒体技术的应用，所面对的数据形式越来越多，数据量也越来越大，相应地，数据管理技术也随之不断地发展进步。在本任务中将学习数据管理技术发展的三个阶段及其主要特征。

二、预备知识

数据管理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。而数据的处理是指对各种数据进行收集、存储、加工和传播和一系列活动的总和。

三、任务实施

总体来说，在应用需求的推动下，在计算机硬件、软件发展的基础上，数据管理技术经历了人工管理、文件系统、数据库系统三个阶段。这三个阶段的特点及其比较如表 1-1 所示。

表 1-1 数据管理技术的三个发展阶段

发展阶段	主要特征
人工管理阶段 (20世纪50年代中期以前)	数据不保存，主要用于科学计算； 程序和数据不可分割，数据不独立； 数据高度冗余且不共享。
文件系统阶段 (20世纪50年代中期到60年代中期)	数据长期保留，可进行查询、修改和删除等操作，大量用于数据处理； 程序与数据分离，数据以文件为单位进行存储，且由操作系统统一管理； 数据的通用性差、冗余度大。
数据库管理系统阶段 (20世纪60年代后期)	对所有数据实施统一管理，数据结构化； 数据与程序之间完全独立； 数据共享性高，冗余度小，易扩充。



子任务三 数据库系统的特点

一、任务描述

了解了有关数据库的基本概念和数据管理技术发展的三个阶段后，在本任务中将学习数据库系统的主要特性。

二、预备知识

数据库的概念最早出现于20世纪60年代，然后得以迅速发展，使用数据库的方式进行数据管理更强调数据的集成性和共享性，可以很好地解决使用文件系统管理数据时遇到的很多问题，降低应用系统的开发周期和维护成本，直到现在仍然是数据管理的主流方式。

三、任务实施

数据库系统主要有以下特性：

(1) 数据的结构化

数据库系统中的每一个数据项之间都是有结构的，数据库以数据模型组织数据，如关系数据库以关系模型来组织数据。

(2) 数据的共享性

数据库系统中的数据文件不是面向某一个应用程序，而是面向整个系统，因此一个数据库不仅可以被一个用户使用，同时也可以被多个用户使用，这样就实现了数据共享，避免了数据库中数据的重复出现。

(3) 数据的独立性

数据库中的数据文件，是面向整个系统的数据文件，部分数据的改变不会影响其他数据的内容和结构。

(4) 数据由DBMS统一管理和控制

数据库的共享是并发的共享，即多个用户可以同时存取数据库中的数据，甚至可以同时存取数据库中的同一个数据。因此DBMS必须提供统一的数据控制功能，包括数据的安全性保护、数据的完整性检查、并发性控制和数据库恢复。



任务二 数据模型

子任务一 数据模型的组成要素

一、任务描述

一般来讲，数据模型是严格定义的一组概念的集合，本任务将介绍数据模型的概念以及它的组成要素。

二、预备知识

模型是现实世界特征的模拟和抽象，数据模型也是一种模型，人们把各个数据对象以及它们之间存在的相互关系的描述称为数据模型。

三、任务实施

数据模型通常由数据结构、数据操作和数据的完整性约束条件三部分组成。

(1) 数据结构

数据结构描述数据库的组成对象以及对象之间的联系，它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，人们通常按照其数据结构的类型来命名数据模型。例如层次结构、网状结构和关系结构的数据模型分别命名为层次模型、网状模型和关系模型。

(2) 数据操作

数据操作是指对数据库中各种对象的实例允许执行的操作的集合，包括操作及有关的操作规则。

数据库主要有查询和更新（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、操作规则以及实现操作的语言等。

(3) 数据的完整性约束条件

数据的完整性约束条件是一组完整性规则。完整性规则是给定的数据模型中数据及其联系所具有制约和依存规则，用以限定符合数据模型的数据库状态以及状态的变化，以保证数据的正确、有效和相容。



子任务二 概念模型

一、任务描述

根据模型应用的不同目的，一般将模型划分为两大类，一类是概念模型，另一类是逻辑模型和物理模型。本任务将介绍概念模型的定义，以及信息世界中涉及到的一些基本概念。

二、预备知识

概念模型用于信息世界的建模，是数据库设计人员进行库设计的有力工具，也是数据库设计人员和用户之间进行交流的语言，因此概念模型也称为信息模型，它是按用户的观点来对数据和信息建模，主要用于数据库设计。

三、任务实施

信息世界主要涉及以下一些概念。

(1) 实体

现实世界客观存在并可以相互区别的事物称为实体。实体可以是具体的人、事、物，也可以是抽象的概念或联系。例如，一个学生、一门课程，一场球赛等都是实体。

(2) 属性

实体所具有的某一特性称为属性。实体可以有多个属性，例如，学生实体可以由“学号”、“姓名”、“性别”、“出生年月”等属性描述。

(3) 码

唯一标识实体的属性集称为码。例如“学号”是学生实体的码。

(4) 实体型

具有相同属性的实体必然具有共同的特征和性质。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、出生年月）就是一个实体型。

(5) 实体集

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。

(6) 联系

在现实世界中，事物内部以及事物之间是有联系的，这些联系反映在信息世界则表示为实体（型）内部的联系和实体（型）之间的联系。实体之间的联系可分为三种：

① 一对一联系：实体集 A 中的一个实体与实体集 B 中至多有一个实体相对应，反之，实体集 B 中的一个实体与实体集 A 中至多有一个实体相对应，则称实体集 A 与实体集 B 有一对一联系。例如，每名学生有一个学号，每个学号对应一名学生，则学生和学号之间具有一对一联系。



②一对多联系：实体集 A 中的一个实体与实体集 B 中的 n 个实体相对应，反之，实体集 B 中的一个实体至多与实体集 A 中的一个实体相对应，则称实体集 A 与实体集 B 有一对多联系。例如，一个班级有若干名学生，每个学生只属于一个班级，则班级与学生之间就具有一对多联系。

③多对多联系：实体集 A 中的每一个实体与实体集 B 中的 n 个实体相对应，反之，实体集 B 中的每一个实体与实体集 A 中的 m 个实体相对应，则称实体集 A 与实体集 B 有多对多联系。例如，一门课程同时有若干名学生选修，一名学生也可以同时选修多门课程，则课程与学生之间具有多对多联系。

子任务三 逻辑模型

一、任务描述

学习了概念模型及信息世界涉及的一些主要概念，本任务将学习什么是逻辑模型。

二、预备知识

逻辑数据模型是对概念数据模型的进一步细化和划分，在概念数据模型定义实体的基础上定义了各个实体的属性，是用户从数据库的角度能够看到的数据的模型，是所使用的数据库管理系统所支持的数据类型。这种数据模型架起了用户和系统之间的桥梁，既要面向用户，同时也考虑到了所用的数据库管理所支持的特性。

三、任务实施

逻辑模型是按计算机系统的观点对数据建模，主要用于数据库管理系统的实现。逻辑数据模型是开发物理数据库的完整文档，主要采用的是层次模型、网状模型、关系模型，其中最常用的是关系模型，对应的数据库称之为关系型数据库，如 MySQL。

逻辑数据模型的内容包括所有的实体、实体的属性、实体之间的关系以及每个实体的主键、实体的外键（用于维护数据完整性）。其主要目标是尽可能详细的描述数据，但是并不涉及这些数据的具体物理实现。

子任务四 物理模型

一、任务描述

逻辑模型和物理模型属于同一类，在本任务中将介绍什么是物理模型。



二、预备知识

物理模型又称物理数据模型，是概念模型和逻辑模型在计算机中的具体表示。该模型描述了数据在物理存储介质上的具体组织，不但与具体的数据库管理系统相关，同时还与具体的操作系统以及硬件有关，但是很多工作都是由DBMS自动完成的。

三、任务实施

物理模型是对数据最底层的抽象，它描述数据在系统内部的表示方式和存在方法，或在磁盘或磁带上的存储方式和存取方法，是面向计算机系统的。物理模型的具体实现是数据库管理系统的任务，数据库设计人员要了解和选择物理模型，最终用户则不必考虑物理级的细节。

总之，物理模型是在逻辑模型的基础上，综合考虑各种存储条件的限制，进行数据库的设计，从而真正实现数据在数据库中的存放。其主要的工作是根据逻辑数据模型中的实体、属性、联系转换成对应的物理模型中的元素，包括定义所有的表和列，定义外键以维持表之间的联系等。

子任务五 层次模型

一、任务描述

数据模型是数据库系统的核心和基础，常见的数据模型有层次模型、网状模型和关系模型，各种数据库管理系统软件是基于某种数据模型的，本任务将介绍层次模型及其特点。

二、预备知识

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），这是1968年IBM公司推出的第一个大型商用数据库管理系统。

三、任务实施

层次模型用树形结构来表示各类实体以及实体间的联系。现实世界中许多实体之间的联系本来就呈现出一种很自然的层次关系，如行政机构、家族关系等。

层次模型像一棵倒置的树，树根、树的分枝点、树叶都称为结点。其结构特点是有且仅有一个结点（根结点）无双亲，其他结点有且仅有一个双亲。

在层次模型中，同一双亲的结点称为兄弟结点，没有子结点的结点称为叶结点，如图1-3所示。

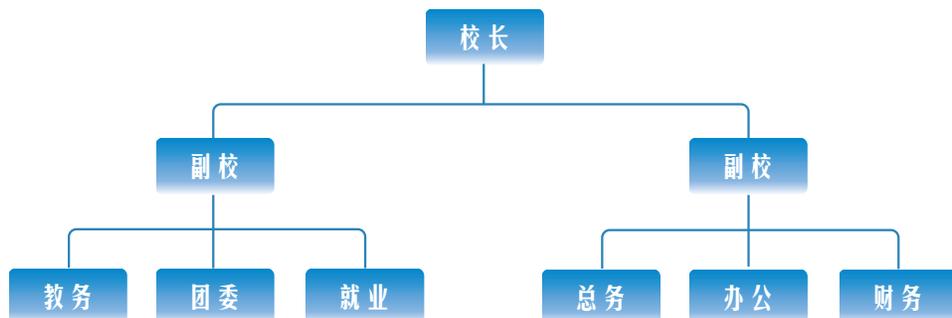


图 1-3 层次模型示意图

子任务六 网状模型

一、任务描述

在现实世界中事物之间的联系更多的是非层次关系的，用层次模型表示非树型结构是很不直接的，网状模型则可以克服这一弊端，本任务将介绍网状模型及其特点。

二、预备知识

网状数据模型的典型代表是DBTG系统，亦称CODASYL系统。这是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（Data Base Task Group, DBTG）提出的一个系统方案。

三、任务实施

网状模型是一种比层次模型更具普遍性的结构，任意一个连通的基本层次模型构成了一个网状模型。网状模型的结构特点是可以有一个以上无双亲，至少有一个结点有一个以上的双亲，结点之间是平等的，不分层次。例如，学校的教师、学生、课程之间的联系是网状模型，如图1-4所示。

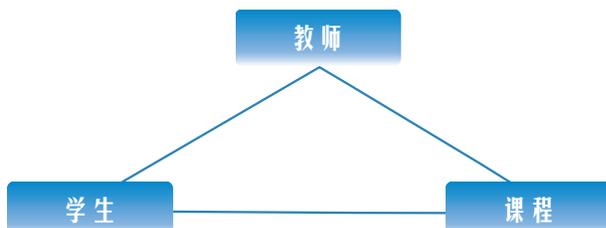


图 1-4 网状模型示意图

层次模型是网状模型的一个特例。



子任务七 关系模型

一、任务描述

关系模型是最重要的一种数据模型，关系数据库系统采用关系模型作为数据的组织方式，本任务将介绍关系模型及其特点。

二、预备知识

关系模型与以往的模型不同，它是建立在严格的数字概念的基础上的，在学习关系模型前先要了解关系模型中的一些术语。

(1) 关系

一个关系对应一张表。

(2) 元组

从表中第二行开始，每一行即为一个元组。

(3) 属性

表中的一列即为一个属性，给每一个属性起一个名称即属性名。

(4) 域

属性的取值范围称为域。例如，学生一门课的成绩取值为0-100，性别的值域则为“男”或“女”。

(5) 关键字

能唯一标识实体集中的各个实体的某个属性或属性集合，称为关键字。例如，在学生实体集中，能作为关键字的属性可以是“学号”，它唯一标识了实体集中的某一个实体，而“姓名”一般不作为关键字，因为可能存在重名。

(6) 主关键字

当关系中有多个属性可以作为关键字而选定其中一个时，则该关键字可称为实体的主关键字。

(7) 外部关键字

在实体的多个属性中，某属性不是该实体的主关键字，却是另一实体的主关键字，称此属性为外部关键字。

(8) 候选关键字

能唯一标识表（关系）中行（元组）的列（属性），则称该属性为候选关键字，也称候选键或候选码。



三、任务实施

在关系模型中数据被组织成一个二维表，这样的表又被称为关系。例如，学籍表、成绩表等。如图1-5所示：

Stuid	Name	Sex	Birtime	depa	Phone
10201	王美丽	女	11/03/2006	电子系	15378760912
20301	周子正	男	03/25/2005	中文系	17823768126
30105	刘世仁	男	11/12/2006	计算机系	17716549218
20902	杨一飞	男	09/28/2006	中文系	15032145566
20604	梁一晓	女	07/08/2006	中文系	13376589123
40502	李昊天	男	03/12/2005	机械系	15324536678
30604	杜炳灿	男	11/26/2005	计算机系	15056721456
10802	秦兰	女	12/12/2006	电子系	15323178922

图 1-5 学生表

构成关系模型的二维表应满足以下条件：

- (1) 表中不允许有重复的字段名，也就是说一个表中不能有两个相同的属性。
- (2) 表中每一列数据必须具有相同的数据类型。
- (3) 表中不允许有两条完全相同的记录。
- (4) 表中行的排列次序以及列的排列次序可以任意，并且其次序不影响表中的关系。

任务三

数据库系统的结构

子任务一 数据库系统模式

一、任务描述

数据库系统的核心是数据库管理系统，它决定着数据组织、存储和管理等方方面面的细节。设计和使用数据库管理系统，首先要了解它的体系结构，确定它的组成部分、以及各部分间相互关系等。本任务将介绍数据库管理系统中模式的基本概念。

二、预备知识

关于数据库的数据需要先理解两个概念：一个是“型”，另一个是“值”。型是指对某一类数据的结构和属性的说明，值是型的一个具体赋值。例如，学生记录定义为(学号，姓名，性别，系别，年龄，籍贯)，这样的形式称为记录型，而(201315130，张晓明，男，计算机系，19，河北省石家庄市)则是该记录型的一个记录值。



三、任务实施

模式是数据库中全体数据的逻辑结构和特征的描述，它仅仅涉及型的描述，不涉及具体的值。模式的一个具体值称为模式的一个实例。同一个模式可以有多个实例。

例如，在学生选课数据库模式中包含学生记录、课程记录和学生选课记录，现有一个具体的学生选课数据库实例，该实例包含了2023年学校中所有学生的记录(如果某校有10000个学生，则有10000个学生记录)、学校开设的所有课程的记录和所有学生选课的记录。

2022年度学生选课数据库模式对应的实例与2023年度学生选课数据库模式对应的实例是不同的。实际上2023年度学生选课数据库的实例也会随时间变化，因为在该年度有的学生可能退学，有的学生可能转系。各个时刻学生选课数据库的实例是不同的、在变化的，不变的是学生选课数据库模式。

相对来说，数据库中型的定义一般比较稳定，变动不大，而值的信息则往往根据实际情况会有一些变动，因为数据库中的数据是在不断更新的。模式反映的是数据的结构及其联系，而实例反映的是数据库某一时刻的状态。

子任务二 数据库系统的三级模式

一、任务描述

在数据库运行的过程中，为了做到数据库的维护及再次开发成本的最小化，一种普遍采用的方式是把数据库进行层次化、模块化分解，目前大多数数据库在进行层次化分解时采用的是三级模式方案。本任务将重点介绍数据库系统的三级模式。

二、预备知识

虽然实际的数据库管理系统产品种类很多，它们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之上，数据的存储结构也各不相同，但它们在体系结构上通常都具有相同的特征，即采用三级模式结构(早期微机上的小型数据库系统除外)并提供两级映像功能。

三、任务实施

数据库系统的三级模式结构是指数据库系统是由外模式、模式和内模式三级构成，如图所示。

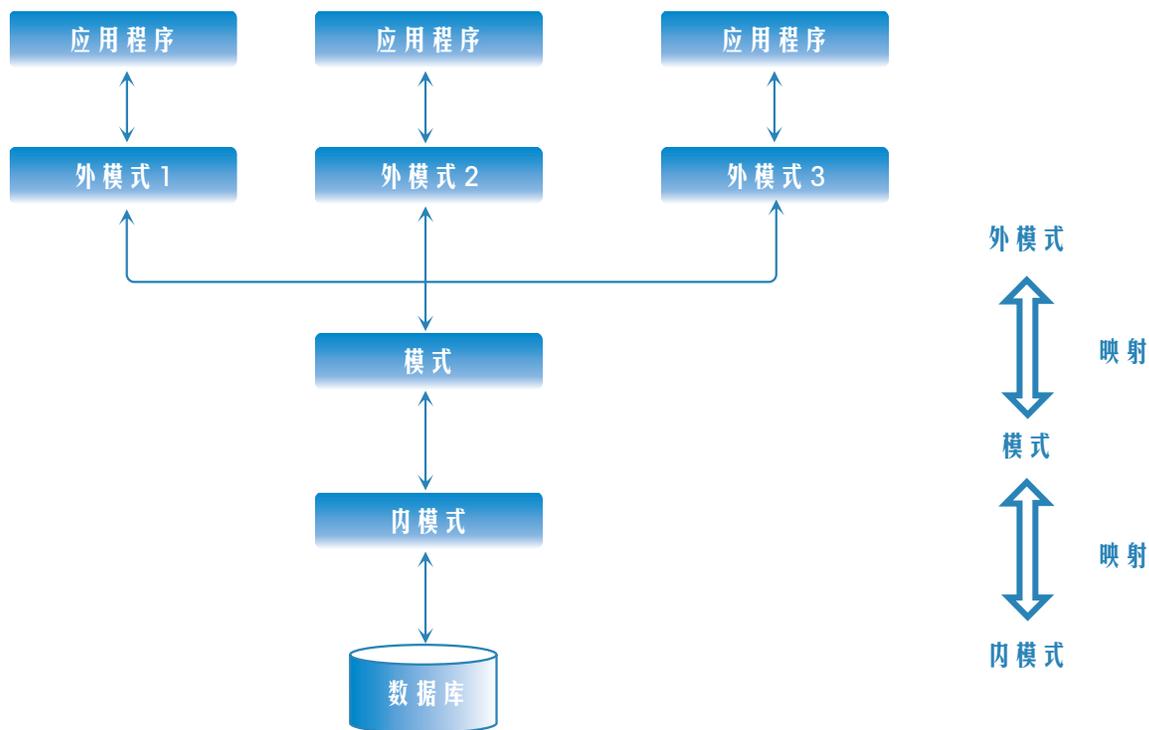


图 1-6 数据库的三级模式和二级映射

(1) 模式

模式也称逻辑模式，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，它是数据库系统模式结构的中间层，既不涉及数据的物理存储细节和硬件环境，又与具体的应用程序、所使用的应用开发工具及高级程序设计语言无关。

模式实际上是数据库数据在逻辑级上的视图。一个数据库只有一个模式。数据库模式以某一种数据模型为基础，统一综合地考虑了所有用户的需求，并将这些需求有机地结合成一个逻辑整体。定义模式时不仅要定义数据的逻辑结构，例如数据记录由哪些数据项构成，数据项的名字、类型、取值范围等；而且要定义数据之间的联系，定义与数据有关的安全性、完整性要求。

(2) 外模式

外模式也称子模式或用户模式，它是数据库用户(包括应用程序员和最终用户)能够看见和使用的局部数据的逻辑结构和特征的描述，是数据库用户的数据视图，是与某一应用有关的数据的逻辑表示。

外模式通常是模式的子集。一个数据库可以有多个外模式。由于它是各个用户的数据视图，如果不同的用户在应用需求、看待数据的方式、对数据保密的要求等方面存在差异，则其外模式描述就是不同的。即使对模式中同一数据，在外模式中的结构、类型、长度、保密级别等都可以不同。另一方面，同一外模式也可以为某一用户的多个应用系统所使用，但一个应用程序只能使用一个外模式。

外模式是保证数据库安全性的一个有力措施。每个用户只能看见和访问所对应的外模式中的数据，数据库中的其余数据是不可见的。



(3) 内模式

内模式也称存储模式，一个数据库只有一个内模式。它是数据物理结构和存储方式的描述，是数据在数据库内部的组织方式。

子任务三 数据库系统的二级映像功能与数据独立性

一、任务描述

外模式、模式和内模式分别对应了数据库数据建模的三个级别，也是数据抽象的三种层次。把数据建模进行分级后各层次的数据功能明确、结构清晰，但是同时也引入了一个新的问题：如何在不同层级的数据之间建立一一对应的关系，这个问题称为各级模式之间的映射问题，本任务将介绍三级数据库模式之间的两级映射及数据独立性。

二、预备知识

数据库系统的三级模式是数据的三个抽象级别，它把数据的具体组织留给数据库管理系统管理，使用户能逻辑地、抽象地处理数据，而不必关心数据在计算机中的具体表示方式与存储方式。为了能够在系统内部实现这三个抽象层次的联系和转换，数据库管理系统在这三级模式之间提供了两层映像：外模式/模式映射和模式/内模式映射。

正是这两层映射保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性。

三、任务实施

(1) 外模式/模式映射

外模式/模式映射解决的是数据库系统的外模式与模式之间的映射对应关系。数据库外模式描述的是现实世界各个方面的直接抽象，相对来说，更为丰富和多样，也难免在不同外模式之间会有冲突和不一致的地方。例如对于长度的描述，外模式1使用厘米为单位，外模式2则使用英寸为单位。

如前所述，模式是全局的，整个数据库只有一个模式，所有外模式都必须映射到一个统一的模式中。因此，全体外模式首先要集成起来，映射为一个完整统一的模式；其次，相互冲突的、不一致的各外模式之间需要规范化，通过规范的协调来构造成为全局模式。经过外模式-模式映射之后，所有多外模式到模式之间就有了确定的对应关系，应用程序或用户对外模式的访问，将被映射到全局统一的模式之中。

(2) 模式/内模式

模式/内模式映射解决的是数据库模式存储到不同存储系统上的问题。数据库内模式指的是数据存储的具体方式。例如，不同数据类型的具体存储方式的定义、数据文件的组织方



式、索引的定义和实现方式等。模式/内模式映射把数据库系统的全局模式映射到物理存储介质上,把对数据的访问转换为对物理存储设备的访问。同一个数据库模式,可能在存储时基于的物理设备不同,导致其模式/内模式映射也不同。对于用户来说,DBMS可以实现多种不同的模式/内模式映射。因此,这种差异对一般用户来说是透明的,使用时仅需关注数据的访问即可,至于访问不同数据需要如何映射到何种物理设备则不需过问。

三级数据库模式之间的两级映射不仅把各层数据模式之间建立了关联关系,而且提供了更好的数据独立性。这个数据独立性体现在两个方面:一个是数据的逻辑独立性;另一个是数据的物理独立性。

数据的逻辑独立性是指当数据库模式发生改变时,数据库外模式不需要随之调整,只需要修改外模式/模式之间的映射关系即可。修改外模式/模式之间的映射关系可以在DBMS帮助下进行,成本比较低,影响范围小。而应用程序都是建立在数据库外模式之上的,如果外模式需要修改,必然会影响整个应用程序的修改,导致大范围的重新开发,增加开发和维护成本。因此,数据的逻辑独立性相当于把数据和应用程序分离开来,通过映射来保证它们之间的联系。数据的逻辑独立性扩大了数据的应用范围、提高了适应性、降低了开发维护成本,具有显著的优点。

数据的物理独立性是指当数据库的物理存储方式发生改变时,也不需要一一修改数据库的模式,仅靠调整模式/内模式映射就可以消除物理存储方式改变带来的差异。由于数据库模式不需要变化,外模式当然也不需要变化,从而应用程序也不用修改。数据的物理独立性进一步提高了数据的适应性和应用范围。数据的逻辑独立性和物理独立性合称数据独立性,这种独立性也是数据库系统最显著的特点之一。

同步训练

简答题

1. 什么是数据库?什么是数据库管理系统?

2. 数据库系统由哪几部分组成?

3. 数据库系统有哪些主要特性?

4. 解释以下概念模型中常用的术语:实体、实体型、实体集。



5. 试述数据模型的分类。

6. 写出数据模型的三个组成要素。

7. 试述层次模型和网状模型的概念。

8. 解释以下术语：关系、属性、域、元组。

9. 试述数据库系统的三级模式结构。

10. 怎样理解数据的物理独立性？

评价反馈

评价项目	学生自评	学生互评	教师评价
1			
2			

项目总结

本项目概述了数据库的基本概念，并通过对数据管理技术进展情况的介绍阐述了数据库技术产生和发展的背景，也说明了数据库系统的优点。

数据模型是数据库系统的核心和基础。本项目简单介绍了概念模型、逻辑模型、层次模型、网状模型和关系模型等数据模型。

本项目还介绍了数据库管理系统内部的系统结构。数据库系统三级模式和两层映像的系统结构保证了数据库系统能够具有较高的逻辑独立性和物理独立性。