

# 第 1 章

## 数据库基础知识

---

数据库 (data base, DB) 作为数据管理领域中的现代技术, 同时也是计算机科学的一个重要分支。本章主要介绍数据库的基础知识, 包括数据库系统 (data base system, DBS) 的基本概论、数据模型、数据库系统模式与结构, 以及数据库系统的组成。

---

### 1.1 数据库系统概论

数据库技术产生于20世纪60年代, 是数据管理的核心技术。数据库管理系统是大型复杂基础软件, 是现代信息系统的核心和基础。

#### 1.1.1 数据库系统的基本概念

数据库系统主要涉及数据、数据库、数据库管理系统 (data base management system, DBMS) 和数据库系统4个基本概念。

- **数据:** 指能输入计算机并能被计算机程序处理的所有符号, 是数据库中存储的基本对象。数据的种类很多, 如数字、文本、图形、图像、音频、视频、学生的档案记录、货物的运输情况等都属于数据。必须赋予一定的含义才能使数据具有意义, 这种含义称为数据的语义, 数据与语义不可分。例如, 63是一个数据, 它可以代表一个学生的某科成绩、某个人的年龄、某系的学生人数等, 只有把63赋予语义后, 才能表示确定的意义。
- **数据库:** 是指在计算机存储设备上建立起来的用于存储数据的仓库, 其中存放的数据是可以长期保留的、有组织的、可共享的数据集合。也就是按照一定的数学模型组织、描述和存储数据, 使得数据库中的数据具有尽可能小的冗余度、较高的数据独立性和易扩展性的特点, 并可在一定范围内共享给多个用户。
- **数据库管理系统:** 是位于用户和操作系统之间的数据管理软件。用它实现数据的定义、组织、存储、管理、操作以及数据库的建立、维护、事务管理、运行管理等功能。

- **数据库系统:**是指带有数据库并利用数据库技术对计算机中的数据进行管理的计算机系统。它可以有组织地、动态地存储大量相关数据，并提供数据处理和信息资源等共享服务。数据库系统一般由满足数据库系统要求的计算机硬件和包括数据库、数据库管理系统、数据库应用开发系统在内的计算机软件，以及数据库系统中的人员组成，如图1.1所示。

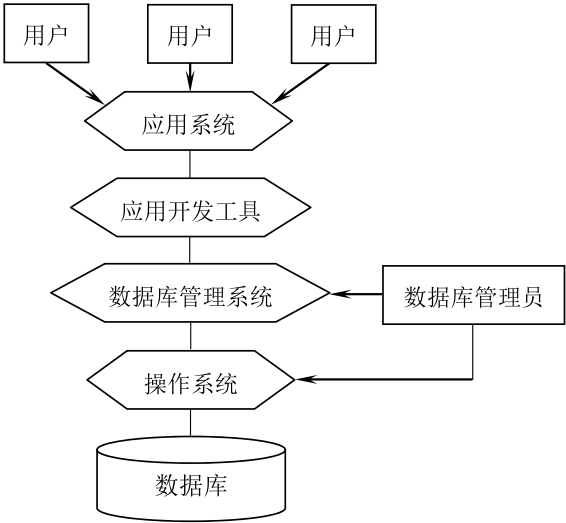


图 1.1 数据库系统

在不引起混淆的情况下，数据库系统也简称为数据库。

1.1.2 数据库技术的产生与特点

截至目前，数据的管理经历了三个阶段：人工管理、文件系统管理和数据库系统管理。

(1) 20世纪50年代中期以前属于人工管理数据的阶段。当时，计算机主要用于科学计算，数据采用批处理的方式，计算机硬件中没有磁盘外部存储设备，软件没有操作系统，因此只能采用人工的方式对数据进行管理。

人工管理数据的特点：数据不方便保存，应用程序管理数据，数据不能共享，数据不具有独立性。人工管理阶段应用程序与数据之间的对应关系如图1.2所示。

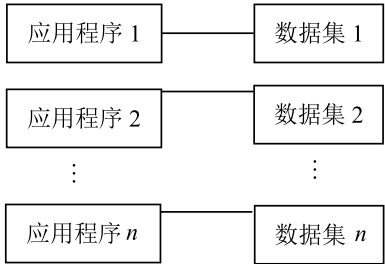


图 1.2 人工管理阶段应用程序与数据之间的对应关系

(2) 20世纪50年代后期到60年代中期属于文件系统阶段。此时，计算机硬件中已经配置了磁盘、磁鼓等外部存储设备，软件操作系统中已经具备专门进行数据管理的系统，即文件系统。

文件系统的特点：数据可以长期保留，有文件系统管理数据，数据的共享性和独立性差、冗余度大。文件系统阶段应用程序与数据之间的对应关系如图1.3所示。

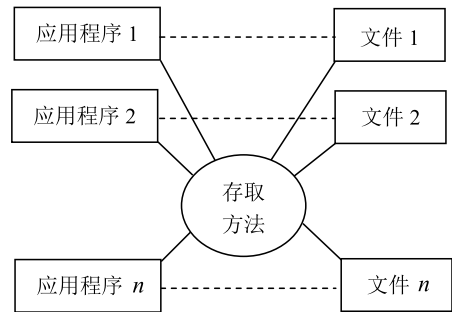


图 1.3 文件系统阶段应用程序与数据之间的对应关系

(3) 从20世纪60年代后期至今属于数据库系统阶段。随着计算机硬件和软件技术的发展，计算机管理对象的规模越来越大，应用范围越来越广，文件系统已经不能满足应用的需求。为了解决多用户、多应用共享数据的问题，使数据尽可能多地为应用服务，一种新的数据管理技术——数据库技术应运而生。此时，专门用于统一管理数据的软件——数据库管理系统成为用户与数据的接口。

数据库系统的特点：数据结构化，数据共享性和独立性高、冗余度低，易扩充，并且数据由数据库管理系统统一管理和控制。

数据库系统阶段应用程序与数据之间的对应关系如图1.4所示。

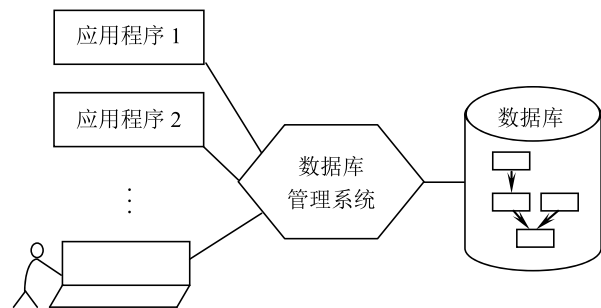


图 1.4 数据库系统阶段应用程序与数据之间的对应关系

## 1.2 数据模型

由于计算机不能直接处理现实世界中的具体事务，因此人们必须事先把要处理的事物特征进行抽象化，转换成计算机能够处理的数据。这个过程使用的工具就是数据模型。从客观世界到计算机世界，包括现实世界→信息世界→计算机世界的抽象过程，这个过程所对应的数据模型分别为概念模型、逻辑模型和物理模型。本节主要介绍数据模型的组成要素和3种不同抽象层次的数据模型（概念模型、逻辑模型和物理模型）等有关内容。

### 1.2.1 数据模型的组成要素

数据模型是对现实世界中某个对象的特征进行的模拟与抽象，是数据库系统的核心和基础。数据模型的严格定义是一组概念的集合。这些概念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。因此，数据模型通常由数据结构、数据操作和完整性约束条件3部分组成。

- 数据结构：是数据对象的集合。它描述数据对象的类型、内容、属性以及数据对象之间的关系，是对系统静态特性的描述。
- 数据操作：是数据库中数据能够执行的操作的集合，包括操作及有关的操作规则，主要有检索（查询）和更新（插入、删除和修改）两类操作，是对系统动态特性的描述。
- 数据完整性约束条件：是数据完整性规则的集合。它是对数据与数据之间的关系制约以及关系依存的规则，用以保证数据的完整性和一致性。

### 1.2.2 数据的概念模型

概念模型是现实世界到计算机世界的第一个中间层次，用于实现现实世界到信息世界的抽象化。它用符号记录现实世界的信息和联系，用规范化的数据库定义语言表示对现实世界的抽象化与描述，与具体的计算机系统无关。概念模型既是数据库设计人员对数据库进行设计的有力工具，也是数据库设计人员与用户交流的有力工具。概念模型涉及如下内容。

#### 1. 概念模型中的基本概念

##### 1) 实体

客观世界存在并可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、一个部门、一门课程、学生的一次选课、部门的一次订货、老师与院系之间的工作关系等都是实体。

##### 2) 属性

实体所具有的某一特性称为属性。一个实体可以由多个属性来刻画。例如，学生实体可以由学号、姓名、性别、出生年月、所在院系、入学时间等属性组成。这些属性组合起来表示一个学生的特征。

##### 3) 码

唯一标识实体的属性集合称为码。例如，学号是学生实体的码。

##### 4) 域

属性的取值范围称为该属性的域，它是具有相同数据类型的数据集合。例如，学号的域为8位整数，姓名域为字符串集合，性别域为{男，女}。

##### 5) 实体型

由于具有相同属性的实体必然具有共同的特征和性质，因此，用实体名及描述实体的各个属性名就完全可以刻画出全部同质实体的共同特征和性质。我们把形式为“实体名（属性名

1, 属性名2, ..., 属性名 $n$ )” 的表示形式称为实体型, 用它刻画实体的共同特征和性质。例如, 学生(学号, 姓名, 性别, 年龄, 所在院系, 入学时间)就是一个实体型, 而(20160016, 李明, 男, 19, 计算机, 2016)是该实体型的一个值。

#### 6) 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。

#### 7) 联系

在现实世界中, 事物内部以及事物之间是有联系的, 这些联系在信息世界中反映为实体型内部的联系和实体型之间的联系。实体型内部的联系通常指组成实体的各个属性之间的联系, 实体型之间的联系通常指不同实体集之间的联系。

### 2. 概念模型中实体型之间的联系

#### 1) 两个实体型之间的联系

两个实体型之间的联系可以分为3种, 即一对一联系、一对多联系和多对多联系。

##### ☉ 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, 就称实体集 $A$ 中的实体型 $A$ 与实体集 $B$ 中的实体型 $B$ 具有一对一联系, 记为1:1。

例如, 在学校的班级实体集和班长实体集中, 一个班级只有一个正班长, 一个班长只在一个班中任职, 班级实体型与班长实体型是一一对一的联系。

##### ☉ 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 $A$ 中至多有一个实体与之联系, 就称实体型 $A$ 与实体型 $B$ 有一对多的联系, 记为1: $n$ 。

例如, 在班级实体集与学生实体集中, 一个班级中有若干名学生, 每个学生只在一个班级中学习, 班级实体型与学生实体型之间就具有一对多的联系。

##### ☉ 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 $A$ 中有 $m$ 个实体( $m \geq 0$ )与之联系, 就称实体型 $A$ 与实体型 $B$ 之间具有多对多的联系, 记为 $m:n$ 。

例如, 在课程实体集和学生实体集中, 一门课程同时有若干个学生选修, 一个学生可以同时选修多门课程, 课程实体型与学生实体型之间就具有多对多的联系。

#### 2) 两个以上实体型之间的联系

两个以上实体型之间也存在一对一、一对多和多对多的联系。

对于 $n$  ( $n > 2$ ) 个实体型  $E_1, E_2, \dots, E_n$ , 若存在实体型  $E_i$  与其余  $n-1$  个实体型  $E_1, \dots, E_{i-1}, E_{i+1}, \dots, E_n$  之间均存在一对一(一对多或多对多)的联系, 而这  $n-1$  个实体型  $E_1, \dots, E_{i-1}, E_{i+1}, \dots, E_n$  之间没有任何联系, 则称 $n$ 个实体型  $E_1, E_2, \dots, E_n$  之间存在一对一(一对多或多对多)的联系。

例如，有课程、教师和参考书3个实体集，如果一门课程可以由若干个教师讲授，使用若干本参考书，而每个教师只讲授一门课程，每一本参考书只供一门课程使用，则课程与教师、参考书之间的联系就是一对多的联系。

又如，有供应商、项目、零件3个实体集，如果一个供应商可以供应多个项目的多种零件，每个项目可以使用多个供应商供应的零件，每种零件可以由不同供应商提供，则供应商、项目和零件之间存在多对多的联系。

### 3) 单个实体型内的联系

同一个实体集内的各个实体之间也可以存在一对一、一对多和多对多的联系。这属于实体型属性之间的联系。例如，职工实体集内部具有领导与被领导的联系，如果某一职工（干部）领导若干名职工，一个职工仅被另一个职工（干部）直接领导，这就是一对多的联系。

显然，一对一联系是一对多联系的特例，而一对多联系是多对多联系的特例。

## 3. 概念模型的E-R图表示方法

概念模型的表示方法有很多，其中最著名、最常用的是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图描述对现实世界进行抽象的概念模型，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。在E-R图中，用矩形表示实体型，矩形内写明实体名称；用椭圆表示属性，并用无向边与相应的实体型相连；用菱形表示联系，菱形内写明联系名，并用无向边分别与有关实体型相连，同时在无向边旁标上联系的类型（1:1、1:n或m:n）。

用E-R图表示两个实体型之间的一对一、一对多和多对多的联系，如图1.5所示。

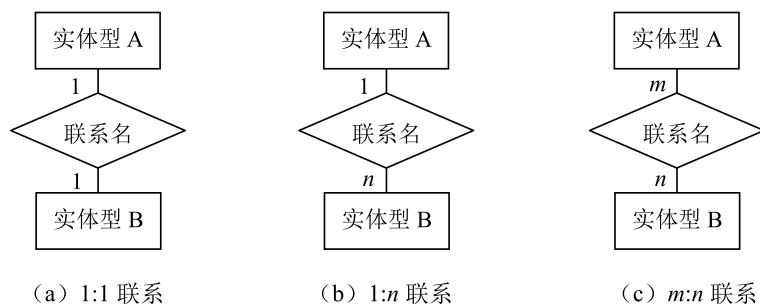


图 1.5 E-R 图

E-R图也可以表示两个以上实体型以及单个实体型内的联系，如课程、教师和参考书3个实体型之间的一对多联系，供应商、项目、零件3个实体型之间的多对多联系，以及职工实体型内部具有领导与被领导的一对多联系，分别如图1.6（a）、图1.6（b）和图1.7所示。

用E-R图表示具有学号、姓名、性别、出生年月、所在院系和入学时间等属性的学生实体，如图1.8所示。

## 4. 一个用E-R图表示概念模型的具体实例

假设有一物资管理处，需要进行物资管理的对象有仓库、零件、供应商、项目和职工，它们是E-R模型中的实体，并具有如下属性：

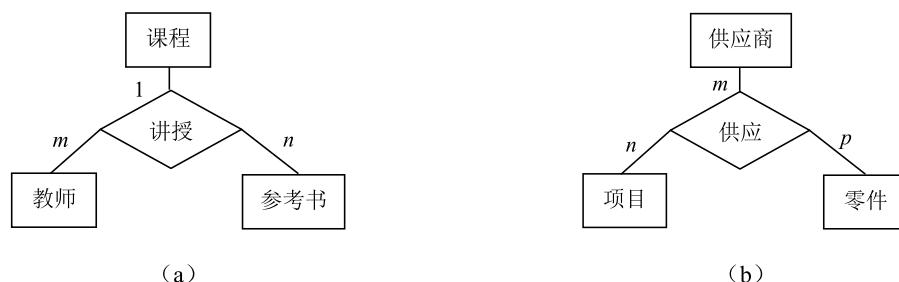


图 1.6 三个实体型之间的联系

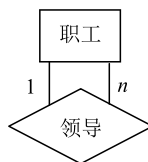


图 1.7 单个实体型之间一对多的联系

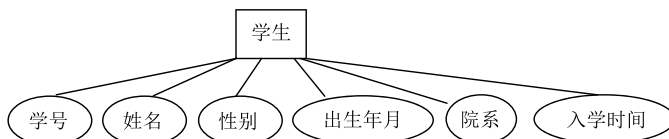


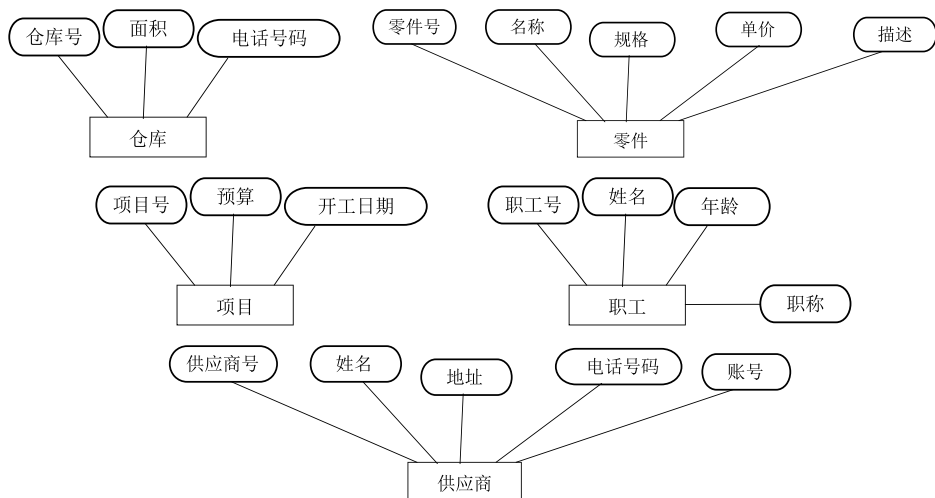
图 1.8 学生实体及属性

- (1) 仓库的属性：包括仓库号、仓库面积、仓库电话号码。
- (2) 零件的属性：包括零件号、零件名称、零件规格、零件单价、零件描述。
- (3) 供应商的属性：包括供应商号、供应商姓名、供应商地址、供应商电话号码、供应商账号。
- (4) 项目的属性：包括项目号、项目预算、开工日期。
- (5) 职工的属性：包括职工号、职工姓名、职工年龄、职称。

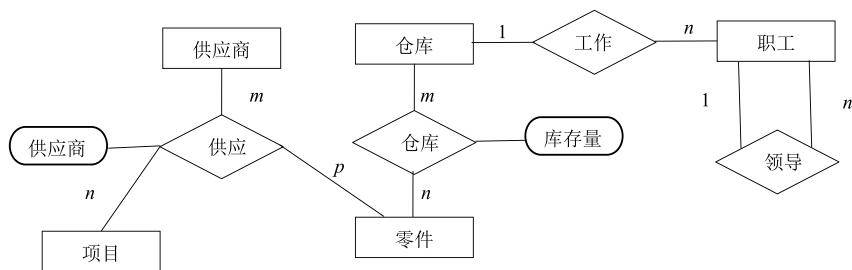
这些实体之间的联系如下：

- 仓库和零件之间具有多对多的联系：因为一个仓库可以存放多种零件，同时一种零件也可以被存放在多个仓库中。现用库存量来表示某种零件在某个仓库中的数量。
- 仓库和职工之间具有一对多的联系：因为在实际工作中，一个仓库可能需要多名仓库管理员（职工），而一名仓库管理员（职工）只能在一个仓库工作。
- 职工实体型中领导与被领导的职工具有一对多的联系：在仓库管理员的职工实体型中，一个仓库只有一名主任，该主任领导若干名管理员，主任与管理员之间具有领导与被领导关系。因此，职工实体型中具有一对多的联系。
- 供应商、项目和零件三者之间具有多对多的联系：因为一个供应商可以为多个项目提供多种零件，每个项目可以使用不同供应商提供的零件，每种零件可由不同供应商供给。因此，供应商、项目和零件三者之间具有多对多的联系。

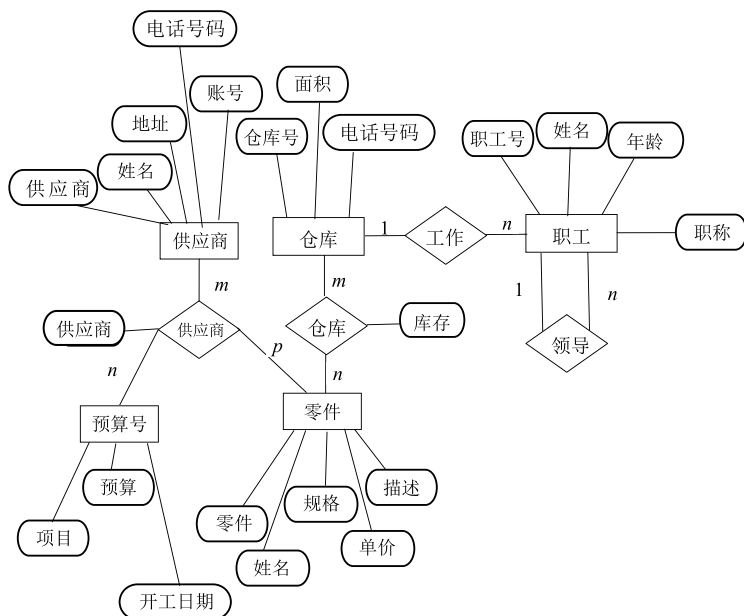
满足上述条件的实体及其属性图如图1.9（a）所示，实体及其联系图如图1.9（b）所示，物资管理的E-R图如图1.9（c）所示。



(a) 实体及其属性图



(b) 实体及其联系图



(c) 完整的实体-联系图

图 1.9 一个用 E-R 图表示概念模型的具体实例



### 1.2.3 数据的逻辑模型

逻辑模型是信息世界到计算机世界的抽象。将信息世界中的概念模型进一步转换成便于计算机处理的数据模型，即为逻辑模型。逻辑模型主要用于DBMS的实现。目前比较成熟地应用在数据库系统中的逻辑模型有层次模型、网状模型和关系模型。它们之间的根本区别在于数据之间联系的表示方式不同（即记录型之间的联系方式不同）。层次模型以“树结构”表示数据之间的联系，网状模型以“图结构”表示数据之间的联系，关系模型用“二维表”（或称为关系）表示数据之间的联系。

#### 1. 层次模型

层次模型是数据库中最早出现的数据模型，它将现实世界的实体之间抽象成一种自上而下的层次关系，用树形结构表示各类实体以及实体间的联系。层次模型的结构特点如下：

- （1）有且只有一个节点没有双亲节点，这个节点称为根节点。
- （2）根以外的其他节点有且只有一个双亲节点。
- （3）上下层节点之间表示一对多的联系。

层次模型本身虽然只能表示一对多的联系，但多对多联系的概念模型可以通过冗余节点法和虚拟节点法分解为一对多的联系，然后使用层次模型来表示。

层次模型的优点是数据结构比较简单、清晰、提供良好的完整性支持，数据库查询效率高。但由于层次模型受文件系统的影响较大，模型受限很多，物理成分复杂，因此不适用于表示非层次性的联系。

#### 2. 网状模型

网状模型是一种非层次模型，它去掉了层次模型的两个限制，与层次模型相比，可以更直接地描述现实世界。网状模型的结构特点如下：

- （1）允许一个以上的节点没有双亲节点。
- （2）一个节点可以有多个双亲节点。
- （3）节点之间表示多对多的联系。

网状模型优于层次模型，具有良好的性能和高效率的存储方式。但其数据结构比较复杂，数据模式和系统实现均不理想。

#### 3. 关系模型

关系模型是目前最重要的一种数据模型。从用户观点看，关系模型由一组关系组成，每个关系的数据结构是一张规范化的二维表。实体与实体之间的联系都用关系来表示。

关系模型的优点如下：

- 建立在严格的数学概念的基础上。
- 关系模型的概念单一，数据结构简单、清晰，用户易懂易用。
- 关系模型的存储路径对用户透明，从而具有更高的数据独立性、更好的安全保密性，简化了程序员的工作和数据库开发建立的工作，但也因此使得查找效率不如层次模型和网状模型。

在此，我们仅对层次模型、网状模型和关系模型的数据结构进行简单的介绍，关于层次模型和网状模型的操作和完整性约束条件的内容可查阅其他参考文献。关系数据模型的详细内容将在本书的第2章介绍。

### 1.2.4 数据的物理模型

物理模型指逻辑模型在计算机中的存储结构。数据库中的数据存储由DBMS完成，它既存储数据，又存储数据之间的联系。层次模型通常采用邻接法和链接法来存储数据及数据之间的联系。网状模型通常采用链接法。关系模型实体与实体之间都用表表示。在关系数据库的物理组织中，有的DBMS中的一张表对应一个操作系统文件，有的DBMS从操作系统中获得若干个大的文件，自己设计表、索引等存储结构。

## 1.3 数据库系统模式与结构

数据库系统根据不同的层次和不同的角度划分为不同的结构。

从用户使用数据库的角度来划分，数据库系统结构分为单用户结构、主从式结构、分布式结构、客户/服务器、浏览器/应用服务器/数据库服务器等多层结构。这种结构称为数据库系统的外部体系结构。

从数据库管理系统的角度来划分，数据库系统通常采用多级模式结构。这种模式结构是数据库系统的一个总体框架，也是数据库管理系统的内部系统结构，能够满足用户方便存储数据和系统高效组织数据的需求。目前，数据库系统采用三级模式和二级映像的系统结构。本节主要介绍数据库系统内部结构这部分内容。

### 1.3.1 数据库系统的三级模式结构

数据库系统的三级模式结构是指数据库系统由外模式、模式和内模式三级构成，如图1.10所示。

#### 1. 模式

模式也称为逻辑模式或概念模式。它是由数据库设计者在统一考虑所有用户需求的基础上，用某种数据模型对数据库中的全部数据的逻辑结构和特征进行的总体描述，是所有用户的公共数据视图。一个数据库只有一种模式，可以用数据库管理系统提供的数据库模式描述语言来定义数据的逻辑结构、数据之间的联系以及与数据有关的安全性和完整性的要求。

在数据模型中有型和值的概念。型是指对某一类数据的结构和属性的描述，值是型的一个具体赋值。模式属于型，模式的一个具体值也称为模式的一个实例。同一个模式可以有很多实例。由于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中的数据在某一时刻的状态，随着数据的更新，实例在不断变化。因此，模式是相对稳定的，实例则是相对变动的。

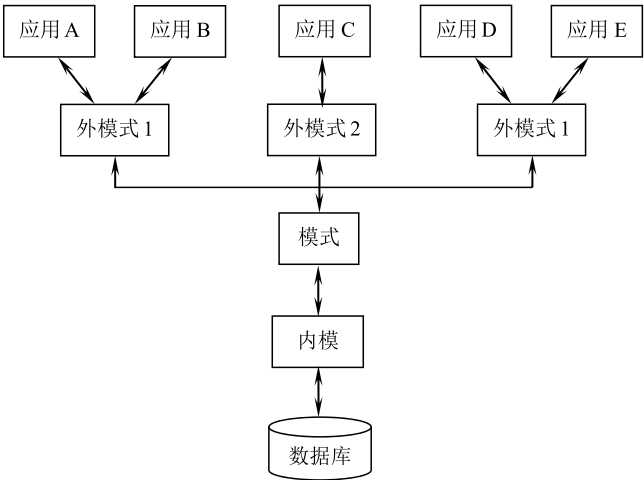


图 1.10 数据库系统的三级模式结构

2. 外模式

外模式也称为子模型或用户模式。它是数据库用户能够看到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，也是与某一应用有关的数据逻辑表示。应用程序的编写依赖于数据的外模式。外模式通常是模式的一个子集，一个数据库可以有多个外模式。一个外模式可以被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。用户可以通过外模式描述语言来描述、定义数据记录（外模式），也可以利用数据操作语言对这些数据记录进行处理。外模式是保证数据库安全性的一个有力措施。

3. 内模式

内模式也称为物理模式或存储模式。它是数据库中全体数据的内部表示或底层描述，描述了数据在存储介质上的存储方式及物理结构，对应着实际存储在外存储介质上的数据库。一个数据库只有一个内模式。内模式由内模式描述语言来描述和定义。

1.3.2 数据库系统的二级映像功能

数据库系统的三级模式是对数据进行抽象的3个级别，为了在数据库系统中实现这3个抽象层次的联系与转换，数据库管理系统在这三级模式之间提供了外模式/模式和模式/内模式的二级映像。这两层映像保证了数据库系统中的数据具有较高的逻辑独立性和物理独立性。

1. 外模式/模式映像

模式描述了数据的全局逻辑结构，外模式描述了数据的局部逻辑结构。对于一个模式，可以定义多个外模式。对于每个外模式，数据库系统都有一个外模式/模式映像，它定义了该外模式与模式之间的对应关系。当模式发生变化时，只要数据库管理员对各个外模式/模式之间的映像做出相应的改变，就可以保持外模式不变，从而对应的应用程序也就无须修改。在数据库中，把用户的应用程序与数据库的逻辑结构相互独立的性质称为数据的逻辑独立性，即当数据的逻辑结构改变时，用户程序也可以不变。

## 2. 模式/内模式映像

数据库中只有一个模式和一个内模式，所以模式/内模式映像是唯一的。模式/内模式的映像定义了数据的全局逻辑结构（模式）与存储结构（内模式）之间的对应关系。当数据的存储结构发生变化时，只要数据库管理员对模式/内模式映射做出相应的改变，就能使模式保持不变，从而应用程序就无须改变。这种存储结构发生变化而应用程序不用改变的性质称为数据的物理独立性。

# 1.4 数据库系统的组成

由1.1.1节可知，数据库系统一般由满足数据库系统要求的计算机硬件和包含数据库、数据库管理系统、数据库应用开发系统等在内的计算机软件以及数据库系统中的人员组成。

## 1.4.1 计算机硬件

在数据库系统中，由于数据库中存放的数据量和DBMS的规模都很大，因此整个数据库系统对硬件资源提出了较高的要求。这些要求是：

- 要有足够大的内存存放操作系统、DBMS核心模块、数据缓冲区和应用程序。
- 要有足够大容量的磁盘或磁盘阵列等设备存放数据库，有足够大容量的磁带（或光盘）做数据备份。
- 系统要有较高的通道能力，以提高数据的传输速率。

满足上述配置的个人计算机、中大型计算机和网络环境下的多台计算机都可以用来支撑数据库系统。

## 1.4.2 计算机软件

数据库系统需要的软件主要包括：

- 建立、使用和维护配置数据库的DBMS。
- 支撑DBMS运行的操作系统。
- 具有与数据库接口的高级语言及其编译系统，便于开发应用程序。
- 以DBMS为核心的为应用开发人员和最终用户提供高效率、多功能的应用程序开发工具。
- 为特定应用环境开发的数据库应用系统。

## 1.4.3 数据库系统中的人员

开发、管理和使用数据库系统的人员包括系统分析员和数据库设计人员、应用程序员、最终用户、数据库管理员（data base administrator, DBA）4类。

- 第1类为系统分析员和数据库设计人员。系统分析员负责应用系统的需求分析和规范说明，他们和用户及数据库管理员一起确定系统的硬件配置，并参与数据库系统的概要设计。数据库设计人员负责数据库中数据的确定和数据库各级模式的设计。
- 第2类为应用程序员，负责编写使用数据库的应用程序。这些应用程序可对数据进行检索、建立、删除或修改。
- 第3类为最终用户，他们利用系统的接口或查询语言访问数据库。
- 第4类是数据库管理员，负责数据库的总体信息控制。DBA的具体职责包括决定数据库中的信息内容和结构；决定数据库的存储结构和存取策略；定义数据库的安全性要求和完整性约束条件；监控数据库的使用和运行；负责数据库的性能改进、数据库的重组和重构，以提高系统的性能。

上述4类不同的人员涉及不同的数据抽象级别，具有不同的数据视图，如图1.11所示。

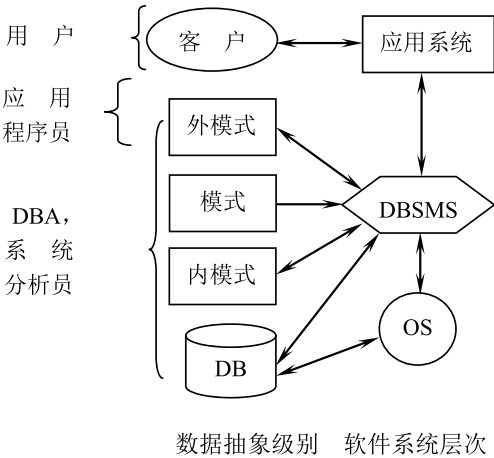


图 1.11 数据库中各类成员的数据视图