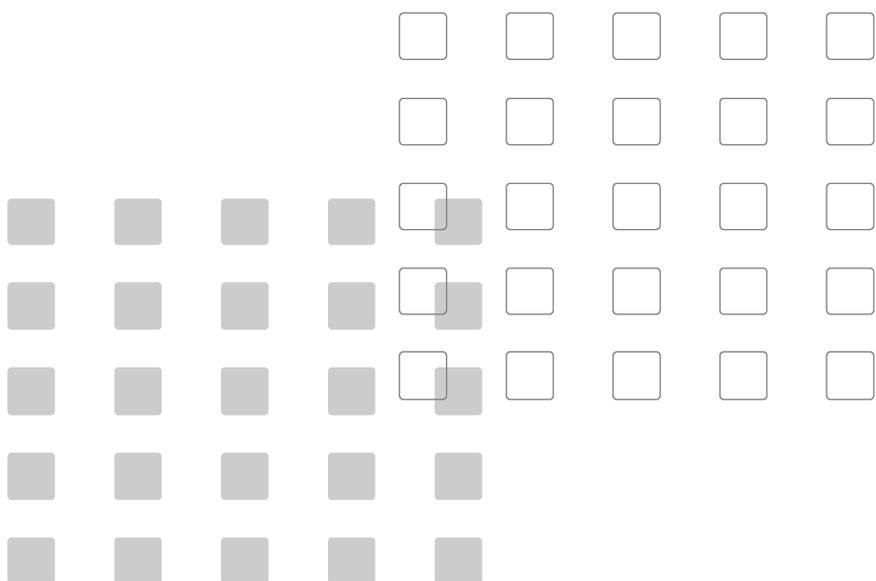


基础原理篇



第 1 章 数据库系统概论

1.1 数据库技术的产生与发展

随着人类社会的不断发展和进步，人们需要处理的数据量越来越大，如何对这些数据进行存储、加工、传输和使用，已日益受到人们的广泛重视。数据库技术就是在这种形势下产生并发展的。

1.1.1 数据管理技术的发展

数据(Data)，即人们用符号对客观事物的描述。数据的种类有很多，包括：文字、图像、声音、图形等。数据是事实或观察的结果，是对客观事物的逻辑归纳，是用于表示客观事物的未经加工的原始素材。

数据是信息的表现形式和载体，可以是符号、文字、数字、语音、图像、视频等。数据和信息是不可分离的，数据是信息的表达，信息是数据的内涵。数据本身没有意义，数据只有对实体行为产生影响时才成为信息。

数据处理的核心问题是数据管理，所谓数据管理，是指对数据的组织、分类、加工、存储、检索和维护。随着计算机软、硬件的不断发展，数据管理经历了如下 3 个发展阶段。

1. 人工管理阶段

20 世纪 50 年代，计算机主要用于科学计算，计算机没有完善的操作系统，没有管理数据的软件，用户以极其原始的方式使用数据，数据不保存，需要时输入，用完删除。在这个阶段，数据面向应用，一组数据对应一个应用程序，致使程序之间存有大量的冗余数据，且易产生数据不一致等问题。

2. 文件系统阶段

20 世纪 50 年代后期到 60 年代中期，计算机技术有了很大的发展，出现了操作系统和管理数据的文件管理系统。在这个阶段，数据不随程序的结束而消失，而是可以长期保存到外存，所需的数据存储在多个不同的文件中，通过编写不同的应用程序来对数据进行检索、修改、插入和删除等操作。但仍然存在数据冗余和不一致，不支持对文件的并发访问，以及难以满足系统安全性要求等弊端，无法满足越来越高的数据处理要求。

3. 数据库系统阶段

20 世纪 60 年代后期，计算机软、硬件技术的飞速发展带来了数据管理的革命，出现了数据管理的新方式——数据库系统(DBS)。数据库系统主要由数据库和数据库管理系统组成，

在数据库系统中，数据以数据库方式存储，使用数据库管理系统可以管理数据库的创建、修改和使用。

与前两种数据管理方式相比，数据库系统具有数据独立性强、冗余小、共享性高、完整性和安全性好等特点。

1.1.2 数据库技术的主要研究领域

数据库技术是使用计算机管理数据的一项新技术，从开始发展到现在，数据库技术已在各行各业得到了广泛应用，是计算机应用的一个重要领域。

数据库是相互关联的数据的集合，但数据库不是简单的数据归集。数据之间包含了一定的逻辑关系，数据库就是根据数据之间的联系和逻辑关系，将数据分门别类地存储。数据库中的数据应具有较小的冗余和较高的数据独立性，可为广大用户所共享。

数据库技术主要应用在需要处理密集型数据的领域，这些领域涉及的数据量大，数据需要长时间保存且为多个应用服务，数据库技术所研究的问题就是如何科学地组织和存储这些数据，以及如何高效地处理和使用这些数据。

1.2 数据库系统的基本概念

1. 数据 (Data)

数据，是指用符号记录下来的可区别的信息。在数据库系统中，数据实际上就是可以被计算机存储、识别的信息。

2. 数据库系统 (Database System, DBS)

数据库系统是数据库技术在计算机中的应用。数据库系统是一个有机结合的人机系统，严格地讲，它是由计算机硬件系统、操作系统、数据库管理系统、数据库、应用程序、数据库管理员和用户组成的。一个数据库系统不仅需要提供提供一个界面，使用户可以方便地建立数据库，快捷地检索和修改数据，还需要提供系统软件来管理存储的数据。数据库系统的组成如图 1-1 所示。

数据库系统必备的特性包括：

- 灵活多样的用户界面；
- 数据的独立性；
- 数据的完整性；
- 查询优化；
- 并发控制；
- 备份与恢复；
- 安全性。

3. 数据库 (Database, DB)

在数据库中，数据与数据的逻辑结构同时存储，各数据文件中数据项的逻辑定义都记录

在“数据字典”中，通过数据库管理系统，用户可以方便地访问数据库中的数据，数据可高度共享。

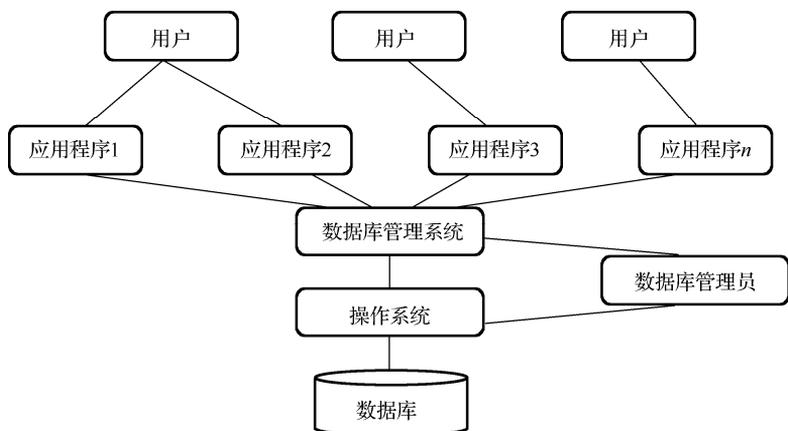


图 1-1 数据库系统组成

4

4. 数据库管理系统(Database Management System, DBMS)

数据库管理系统是数据库系统的核心，在操作系统的支持下，其对内负责管理数据库中的数据，对外负责对用户提供操作数据库的界面。数据库管理系统的主要功能如下。

(1) 数据定义

DBMS 提供了数据定义语言(Data Definition Language, DDL)，用于定义数据库中数据的逻辑结构。

(2) 数据操纵

DBMS 提供了数据操纵语言(Data Manipulation Language, DML)，用于对数据库进行检索、插入、修改和删除等基本操作。数据操纵语言一般分为两类：一类为自主型，一类为宿主型。自主型可独立使用，无须依赖其他程序设计语言；而宿主型则需嵌入到其他程序设计语言(如 C 语言等)中。

(3) 数据库运行控制

DBMS 提供了运行控制机制，包括：数据完整性控制、并发控制、安全性控制及数据备份和恢复功能。

5. 数据库管理员(DataBase Administrator, DBA)

数据库管理员不仅要熟悉数据库管理软件的使用，还应熟悉本行业的业务工作，其主要职责是：管理用户对数据库及相关软件的使用，对数据库进行维护，确保数据库正常运行。

6. 用户(User)

用户，即数据库的使用者，不同的用户可通过不同的方式访问数据库，既可通过良好的用户界面访问数据库，也可使用数据库语言直接访问，但必须是已经授权的用户，不同的用户被授予的访问权限也可能不同。

1.3 数据模型

1.3.1 现实世界的抽象过程

现实世界是指实际存在的事物或现象。各种事物都有着自己的许多特性，在众多的事物之间，它们又存在着千丝万缕的联系。

现实存在的事物，例如：桌子、人，桌子有高有低、有方有圆、有黄有红等；人有男有女、有胖有瘦、有白有黑等。这些都是事物自身拥有的特性，这些事物用计算机是无法直接处理的，只有将这些事物的特性数据化以后，才能被计算机所接受，才能被计算机处理。但是如何将现实世界的这些事物转换成计算机所能处理的数据，也就是如何将这些事物的特性及事物之间的联系转换成数据，就是本节要讨论的现实世界的抽象过程。

现实事物是不可能自动转换成计算机所能处理的数据的，它必须通过人的帮助才能转换。首先，应是对现实世界的事物有了发现，这种发现通过人们的头脑反应、理解后，转换成信息，然后将这些在人的头脑中反应的信息转换成计算机所能处理的数据。一般来说，我们把现实世界实际存在的东西称为事物，每件事物都有其基本特征，现实世界中的事物在人脑中的反应称为信息，这些信息被具体描述成一个个实体，这些实体就对应于现实世界的一件件事物，而事物的特征即被描述成实体的属性，再把信息在计算机中的物理表示称为数据，对应于实体和属性，在数据世界中称为记录和数据项。现实世界的抽象过程见图 1-2。

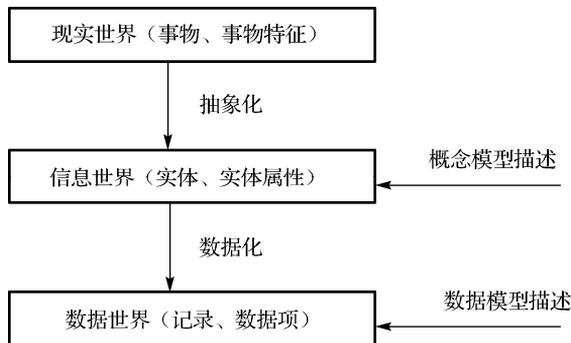


图 1-2 现实世界的抽象过程

1.3.2 概念模型

信息世界是现实世界转换到数据世界(又称机器世界)的中间环节，是人们对现实世界的认识和理解。信息世界用概念模型描述，概念模型不依赖于具体的机器世界，而是与现实世界紧密联系。要进行数据库设计，首先必须给出概念模型，概念模型能很好地体现出设计人员的思想，且设计简单，有利于设计人员与用户交流。

1. 基本概念

(1) 实体(Entity)

实体即客观存在且可区别的事物在信息世界的反映，实体既可以是实际的事物，又可以是一种概念或现象，例如，一个教师、一本书、一堂课、一个程序等都可称为实体。

(2) 实体集(Entity Set)

具有相同属性名，而属性值可有所不同的实体的集合即为实体集。在实体集中，不能存在两个或两个以上相同的实体，如学校的全体教工、书店的全部书籍、工厂的所有设备等都是实体集。为了区别不同的实体集，应给每个实体集取一个名字，称为实体名。

(3) 实体型(Entity Type)

实体型即实体集的命名表示，由实体名和实体集的各属性名构成。如教工登记表(编号，姓名，性别，年龄，婚否，职称，部门)就是全体教工实体集的实体型。

(4) 属性(Attribute)

属性即事物具有的具体特征，在实体中称为属性，实体是由若干个属性来描述的。如教工实体是由编号、姓名、性别、年龄、婚否、职称、部门这些属性来描述的。

(5) 域(Domain)

某个属性的取值范围称该属性的域。如性别的域为“男”和“女”，姓名的域取8个字节长的字符串，职称的域定义为“教授”、“副教授”、“讲师”、“助教”等。域限制属性的取值。

(6) 键(Key)

在实体集中，不允许完全相同的两个实体存在，即在同一个实体集中的实体，相互间至少应有一个属性(或属性组)的值不同，也就是应有一个能唯一区分一个实体的属性或属性组存在，该属性或属性组就称为键，也称为码。如教工实体中，编号就可作为键，因为每个编号只对应一个教工实体。

(7) 联系(Relationship)

现实世界中的事物存在着联系，这种联系反映在概念模型中就表现为实体集本身内部的联系和实体集间外部的联系。实体集的内部联系表现在组成实体的各属性之间，如姓名与职称之间是“拥有”关系；实体集的外部联系表现为不同实体集之间，如教师实体与学生实体之间是“教学”关系。联系一般也有联系名。

2. 实体集间的联系

两个实体集间的联系一般分为3类。

(1) 一对一联系(1:1)

假设有两个实体集 A 和 B ，如果实体集 A 中的每个实体至多与实体集 B 中的一个实体相联系，而实体集 B 中的每个实体也至多与实体集 A 中的一个实体相联系，则称实体集 A 与实体集 B 或实体集 B 与实体集 A 是一对一联系，一般可记为 $1:1$ ，如图 1-3 所示。

例如，一个部门只有一个主任，而一个主任也只在在一个部门任职。则可认为主任与部门之间是一对一联系。

(2) 一对多联系(1:n)

假设有两个实体集 A 和 B ，如果实体集 A 中的每个实体都可以与实体集 B 中的多个实体

相联系，而实体集 B 中的每个实体却至多只能够与实体集 A 中的一个实体相联系，则称实体集 A 与实体集 B 是一对多联系，一般可记为 $1:n$ 。例如，一个班级可以有 multiple 学生，而每个学生只能在一个班级中。则可认为班级与学生之间是一对多联系，如图 1-4 所示。

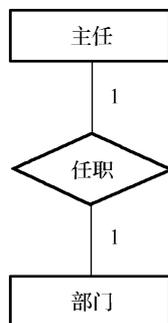
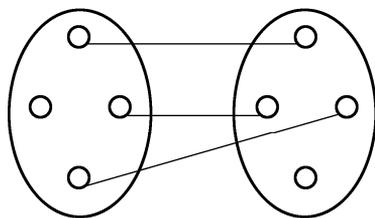


图 1-3 两个实体集间的一对一联系

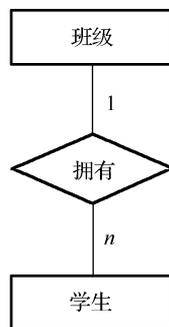
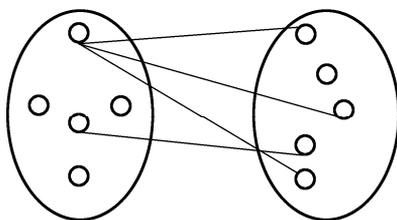


图 1-4 两个实体集间的一对多联系

(3) 多对多联系 ($m:n$)

假设有两个实体集 A 和 B ，如果实体集 A 中的每个实体都能够与实体集 B 中的多个实体相联系，而实体集 B 中的每个实体也能够与实体集 A 中的多个实体相联系，则称实体集 A 与实体集 B 或实体集 B 与实体集 A 是多对多联系，一般可记为 $m:n$ 。例如，一个教师可以教多门课程，一门课也可由多个教师任教。则可认为教师与课程之间是多对多联系。如图 1-5 所示。

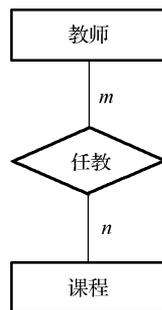
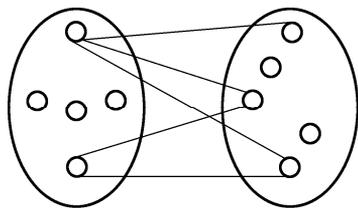


图 1-5 两个实体集间的多对多联系

1.3.3 数据模型

数据模型是指数据在数据库中的存储结构，任何一种数据库管理系统都支持一种数据模型。较为常见的数据模型有：层次数据模型、网状数据模型、关系数据模型。近年来还出现了一种新的数据模型——面向对象数据模型。

1. 层次数据模型(Hierarchical Model)

层次数据模型是最早使用的一种数据模型，采用层次模型的数据库通过链接方式将相互关联的记录组织起来，形成一种层次关系，构成一种树形结构。这种树形结构，就像一棵倒挂的树，若把每条记录看成是树上的一个节点，则最上一层的节点类似于树根，称为根节点。结构中的每个节点都可以链接一个或多个节点，这些节点称为后继节点或子节点。与子节点相链接的上一层节点称为该子节点的前驱节点或父节点，链接则表示节点之间的联系。

层次数据模型的特点包括：

- 有且仅有一个节点没有父节点(即根节点)，除根节点以外，其他节点有且只有一个父节点；
- 每个节点都可有一个或多个子节点。

现实生活中有很多按层次结构组织数据的实例，如行政管理、目录管理、族谱等。图 1-6 为某学院行政管理的数据模型。

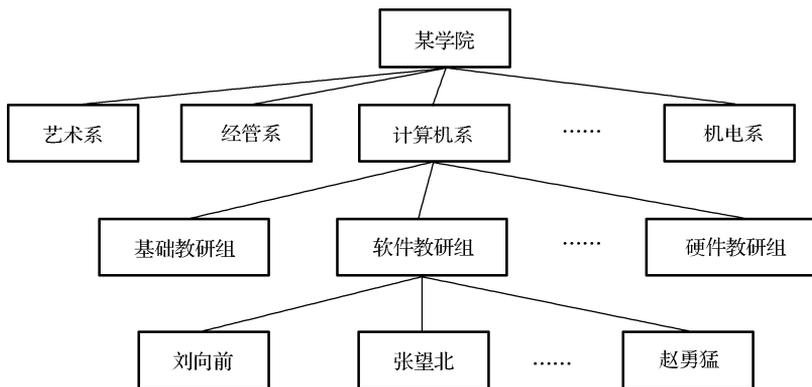


图 1-6 某学院行政管理的数据模型

由图 1-6 可知，该学院设有若干个系，各系设有若干个教研组，各教研组又拥有若干名教师，因此形成一种层次关系，同一层次的节点称为兄弟节点，没有子节点的节点称为叶子节点。

2. 网状数据模型(Network Model)

网状数据模型也是早期经常使用的一种数据模型，网状数据模型采用网状结构表示实体和实体之间的联系，网状数据库则是使用网络模型作为自己的存储结构。在此结构中，各记录便组成网络中的节点，有联系的各节点通过链接的方式联系在一起，构成一个网状结构，即图形结构，这种结构节点之间的联系较为复杂。

网状数据模型的特点包括：

- 允许多个节点没有父节点，允许节点有多个父节点；

- 允许两个节点之间有多种联系。

例如，在某学院教学管理中，系、教师、专业、学生之间的联系就构成一个网状结构，如图 1-7 所示。

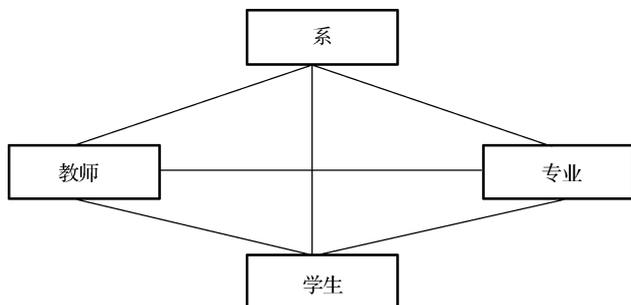


图 1-7 某学院教学管理的数据模型

由图 1-7 可知，一个系可以拥有教师、学生和专业的；一个教师可以属于一个系、可以教若干学生、可以任教某个专业；一个学生又可属于某个系、可以由某些教师教课、可以学某个专业；一个专业可以被设在某个系、可以由某些教师任教、可以被某些学生选修。它们之间就构成一个复杂的网状结构。

3. 关系数据模型 (Relational Model)

关系数据模型是目前使用最广泛的一种数据模型，关系数据模型以其概念简单清晰、操作直观方便、易学易用等优势，受到了众多用户的青睐。现在 90%以上的数据库产品都是以关系模型为基础的。关系数据模型采用关系作为逻辑结构，关系实际上就是一张二维表，一般简称表。

关系数据模型的特点包括：

- 每张二维表都是由行和列构成的，每行称为一条记录(或一个元组)，每列称为一个字段(或一个属性)；
- 关系模型中，实体及实体间的联系都用关系来表示，其操作对象和操作结果都是关系。

例如，学院的教工管理、学生学籍管理等一般都采用关系数据模型(即使用表)，如表 1-1、表 1-2 所示。

表 1-1 教工登记表

教师编号	姓名	性别	年龄	婚否	职称	基本工资	部门
JSJ001	江河	男	30	1	讲师	880.00	计算机系
JSJ002	张大伟	男	24	0	助教	660.00	计算机系
JGX001	王冠	男	32	1	讲师	800.00	经管系
JGX002	刘柳	女	38	1	副教授	1000.00	经管系
JCB002	张扬	女	28	0	讲师	800.00	基础部
JGX003	王芝环	女	24	0	助教	500.00	经管系
JCB001	汪洋	男	27	1	NULL	NULL	基础部

表 1-2 教工工资表

工 资 号	姓 名	基本工资	岗位补贴	奖 金	扣 除	实发工资
1	江河	880.00	400.00	400.00	250.00	1430.00
2	张大伟	660.00	300.00	250.00	120.00	1090.00
3	王冠	800.00	400.00	300.00	200.00	1300.00
4	刘柳	1000.00	600.00	400.00	260.00	1740.00
5	张扬	800.00	400.00	300.00	180.00	1320.00
6	王芝环	500.00	300.00	150.00	150.00	800.00
7	李力	900.00	600.00	400.00	236.00	1664.00

关系模型数据库操作方便、便于管理，是目前使用最多的一种数据模型。

4. 面向对象数据模型(Object-Oriented Model)

前三种数据模型所支持的数据类型有限，不能实现对诸如声、像、画、影视等数据的处理，面向对象数据模型是随着数据库技术的飞速发展应运而生的一种新型的数据模型。面向对象模型以对象为基本结构，而面向对象数据库是数据库技术与面向对象技术相结合的产物，面向对象数据库系统能够有效地处理计算机辅助设计、办公自动化、多媒体应用等方面的数据库应用，是目前数据库技术的热点研究方向。不过，与关系数据库相比，面向对象数据库技术与理论还不是很成熟。

10

1.4 数据库体系结构

数据库的体系结构分为 3 层：数据库的物理结构、数据库的逻辑结构、数据库的视图结构。数据库的物理结构是数据的物理存储方式，一般称为内模式；数据库的视图结构是用户的数据视图，最接近用户，一般称为外模式；数据库的逻辑结构介于物理结构和视图结构两者之间，一般称为模式。对于一个数据库系统来说，只有一个内模式，也只有一个模式，但可有多个外模式，如图 1-8 所示。

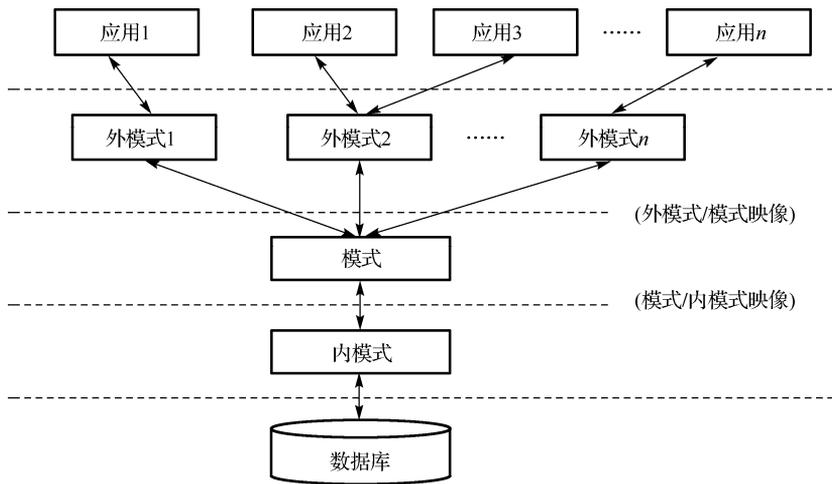


图 1-8 数据库体系结构

1. 模式

模式 (Schema) 是数据库的逻辑结构。模式又称为逻辑模式 (Logic Schema) 或概念模式 (Conceptual Schema), 表示数据库的全部信息内容, 定义数据库全部数据的逻辑结构, 其形式比数据的物理结构抽象些, 主要描述数据库中存储什么数据, 以及这些数据之间有何种关系。

2. 外模式

外模式 (External Schema) 是数据库的视图结构。外模式又称为子模式或用户模式, 是最接近用户的模式, 是模式的子集, 即从模式中抽取的部分或全部, 对应于不同的用户。用户的应用目的不同、使用权限不同, 对应的外模式的定义就不同, 每个用户只能使用自己权限范围内能调用的外模式的数据, 而无法涉及其他外模式的数据。

3. 内模式

内模式 (Internal Schema) 是数据库的物理结构。内模式又称为存储模式 (Storage Schema) 或物理模式 (Physical Schema), 是整个数据库的最底层表示, 用于定义数据的存储方式和物理结构。

4. 映像

数据库体系结构中, 还定义了二级映像 (Mapping)——模式/内模式间的映像和外模式/模式间的映像, 以保证数据库系统的数据具有较高的独立性。下面介绍与映像相关的几个概念。

数据独立性: 是指当修改某一层级的模式定义时, 不至于影响其上一层级模式定义的能力。数据独立性包括两个层次: 一是物理独立性, 二是逻辑独立性。

物理独立性: 是指用户的应用程序与存储在数据库中的数据是相互独立的, 应用程序处理的是数据的逻辑结构, 至于存储文件中的数据在磁盘中如何存储, 用户无须了解, 当存储文件中的数据在磁盘中的存储位置发生改变时, 应用程序无须发生改变。

也就是说, 应用程序不依赖于数据库中存放数据的物理结构, 我们可以对存储的数据进行修改, 而不必去改动应用程序。例如, 原有数据是按一种标准顺序存储的, 若要改用另一种标准进行存储, 则对物理数据的改变并不会影响到现有的数据库的逻辑结构以及数据库的应用程序。

逻辑独立性: 是指用户的应用程序与数据库定义的逻辑结构是相互独立的, 当数据库的逻辑结构发生变化时, 不至于影响到用户的应用程序。

也就是说, 我们可以单独对数据库的逻辑结构进行修改, 而不必修改使用数据库的应用程序。例如, 数据库中有一“教师登记表”, 我们要给该表添加一个字段“职务”, 只需对表的逻辑结构进行修改, 要用到这个表的应用程序都无须改动。

模式/内模式映像: 定义了模式与内模式之间的对应关系, 当数据库的存储结构发生改变时, 只要改变相应的模式/内模式映像, 就可使模式保持不变, 从而使外模式也保持不变。模式/内模式映像是保持数据物理独立性的关键。

外模式/模式映像: 定义了外模式与模式之间的对应关系, 当模式发生改变时, 只要改

变相应的外模式/模式映像，可使外模式保持不变。外模式/模式映像是实现数据逻辑独立性的关键。

1.5 小结

本章主要介绍了数据库系统的基本概念，主要内容如下。

(1) 数据管理技术发展的 3 个阶段：人工管理阶段、文件系统阶段、数据库系统阶段。

(2) 数据库系统的组成：由计算机硬件系统、操作系统、数据库管理系统、数据库、应用程序、数据库管理员和用户组成。

(3) 现实世界的抽象过程：现实世界(事物、事物特征)→信息世界(实体、实体属性)→数据世界(记录、数据项)。

(4) 概念模型的基本概念：实体、实体集、实体型、属性、域、键、联系等。

(5) 实体间的联系类型：一对一、一对多、多对多。

(6) 常见的数据模型：层次模型、网状模型、关系模型。

(7) 数据库系统的三级模式结构及二级映像：模式、外模式、内模式及外模式/模式映像和模式/内模式映像。

12

1.6 习题

一、填空题

1. 数据管理技术经历了_____、_____、_____三个阶段。
2. DBMS 是指_____，它是位于_____和_____之间的一层管理软件。
3. 实体间的联系一般分为三类：_____、_____、_____。
4. 数据库具有数据结构化和最小的_____及较高的_____等特点。
5. 每个关系实际上就是一张_____。

二、选择题

1. 数据库管理系统可实现对数据库中的数据进行查询、插入、修改和删除，这类功能称为()。

A. 数据定义功能	B. 数据操纵功能
C. 数据控制功能	D. 数据管理功能
2. 要保证数据库的数据独立性，需要修改的是()。

A. 模式与外模式	B. 模式与内模式
C. 三级模式间的二级映像	D. 模式、外模式、内模式
3. 在数据库的三级模式结构中，用于描述数据库中全体数据的全局逻辑结构和特性的是()。

A. 外模式	B. 模式	C. 内模式	D. 视图模式
--------	-------	--------	---------

