

第1章 数据库基础

“数据库”一词起源于20世纪50年代初，当时美国为了战争的需要，把各种情报集中在一起，存储在计算机里，称为 Information Base 或 DataBase。在20世纪60年代的“软件危机”中，数据库技术作为软件学科的一个分支应运而生。

随着计算机与网络的普及，数字化生活给人类生活和工作方式带来了巨大的变化，而支撑实现数字化的关键技术之一就是数据库系统。现在，数据库系统已在当代社会生活中获得了广泛的应用。当我们在QQ上聊天、微博上留言、网上购物、ATM机上存取款、超市购物付款时都在享受着数据库系统的服务。

1.1 数据库基础知识

1.1.1 数据库系统的基本概念

1. 信息和数据

描述事物的符号记录称为数据。它是数据库中存储的基本对象。数据有数字、文字、图形、图像、动画、影像、声音等多种表现形式，它们都可以经过数字化后存入计算机。凡是计算机中用来描述事物特征的记录，都可以统称为数据。

信息是现实世界事物的存在方式或运动状态的反映；或认为，信息是一种已经被加工为特定形式的数据。信息是人们消化理解了的数据，是对客观世界的认识，即知识。人们通过解释、推理、归纳、分析和综合等方法，从数据所获得的有意义的内容称为信息。因此，数据是信息存在的一种形式，只有通过解释或处理的数据才能成为有用的信息。

数据与信息既有联系又有区别，数据是载荷信息的物理符号或称载体，用于描述事物，传递或表示信息。信息是抽象的，不随数据形式而改变。信息是反映客观现实世界的知识，用不同的数据形式可以表示同样的信息。

如“一名女大学生”，这种描述是一般意义的信息。为了在计算机中存储和处理这个对象，必须提取她的属性和特征，根据需要，往往只提取部分必要的特征，可以从“姓名、性别、出生日期、政治面貌、班级编号、照片”等属性来加以描述，具体形式如(史晓庆，女，1991-4-3，中共党员，201001，登记照)。

2. 数据库

数据库(Database，简称DB)可以理解为“存放数据的仓库”，只不过这个仓库是计算机的存储设备。所谓数据库是指长期存储在计算机内的、有组织的、可共享的数据集合。数据库中的数据按一定的数据模型描述、组织和存储，具有较小的冗余度、较高的数据独立性和易扩展性，并可为用户共享。

在经济管理的日常工作中，常常需要把某些相关的数据放进这样的“仓库”，并根据管理的需要进行相应的处理。

例如,企业或事业单位的人事部门常常要把本单位职工的基本情况(职工号、姓名、年龄、性别、籍贯、工资、简历等)存放在表中,这张表就可以看成一个数据库。有了这个“数据仓库”我们就可以根据需要随时查询某职工的基本情况,也可以查询工资在某个范围内的职工人数等等。这些工作如果都能在计算机上自动进行,那我们的人事管理就可以达到极高的水平。此外,在财务管理、仓库管理、生产管理中也需要建立众多的这种“数据库”,使其可以利用计算机实现财务、仓库、生产的自动化管理。

再如,学校把学生的基本情况(如学号、姓名、性别、出生日期、政治面貌、照片)存放在“学生”表中,把课程信息(如课程编号、课程名称、课程类别、学分)存放在“课程”表中,把成绩信息(包括学号、课程编号、分数)存放在“成绩”表中,这三个表能组成一个最简单的数据库,可以根据需要查询该库的某个学生的基本情况、学生选课及成绩情况等。

3. 数据库管理系统

数据库管理系统(Database Management System, DBMS)是一种用于管理数据库的计算机系统软件,是位于用户与操作系统之间的一层数据管理软件,是数据库系统的一个重要组成部分。用户通过 DBMS 访问数据库中的数据,数据库管理员也通过 DBMS 进行数据库的维护工作。它可使多个应用程序和用户通过不同的方法在同时或不同时刻去建立、修改和询问数据库。其功能包括:

➤ 数据定义功能

DBMS 提供数据定义语言 DDL(Data Definition Language),供用户定义数据库的三级模式结构、两级映像以及完整性约束和保密限制等约束。DDL 主要用于建立、修改数据库的库结构。

➤ 数据操纵功能

DBMS 提供数据操作语言 DML(Data Manipulation Language),供用户实现对数据的追加、删除、更新、查询等操作。

➤ 数据库的运行管理

数据库的运行管理功能是 DBMS 的运行控制、管理功能,包括多用户环境下的并发控制、安全性检查和存取限制控制、完整性检查和执行、运行日志的组织管理、事务的管理和自动恢复,保证了数据库系统的正常运行。

➤ 数据组织、存储与管理

DBMS 要分类组织、存储和管理各种数据,包括数据字典、用户数据、存取路径等,须确定以何种文件结构和存取方式在存储级上组织这些数据,如何实现数据之间的联系。数据组织和存储的基本目标是提高存储空间利用率,选择合适的存取方法提高存取效率。

➤ 数据库的维护

这部分包括数据库的数据载入、转换、转储、数据库的重组和重构以及性能监控等功能,这些功能分别由各个使用程序来完成。

➤ 数据库的通信

DBMS 具有与操作系统的联机处理、分时系统及远程作业输入的相关接口,负责处理数据的传送,与其他软件系统进行通信。

数据库管理系统一般提供如下几种相应的数据语言:

(1) 数据定义语言(DDL),负责数据的模式定义与数据的物理存取构建。

(2) 数据操纵语言(DML)，实现对数据库数据的基本存取操作：检索、插入、修改和删除等。

(3) 数据控制语言(DCL)，负责数据的安全性、完整性和并发控制等，对数据库运行进行有效的控制和管理，以确保数据正确有效。

DBMS 不仅具有最基本的数据管理功能，还能保证数据的完整性、安全性，提供多用户的并发控制，当数据库出现故障时对系统进行恢复。

4. 数据库系统人员

数据库用户是管理、开发、使用数据库的主体。根据工作任务的差异，数据库用户通常可以分成终端用户、应用程序员和数据库管理员3种类型。

(1) 终端用户

终端用户是应用程序的使用者，使用数据库系统提供的终端命令语言，或者菜单驱动、表格驱动、图形显示和报表生成等对话方式，来存取和应用数据库中的数据。这类人员是一些并不精通计算机和程序设计的各级管理人员，但必须接受必要的数据库应用培训。

(2) 应用程序员

应用程序员是负责设计和编制应用程序的人员。他们使用高级语言设计和编写应用程序，供终端用户使用。应用程序员不仅要求具有较高的技术专长，而且还要具备较深的资历，熟悉部门全部数据的性质和用途，兼有系统程序员、系统分析员的能力。其具体职责：①决定数据库的内容和结构；②决定数据库的存储结构和存取策略，使数据的存储空间利用率和存取效率均较优；③定义数据的安全性要求和完整性约束条件；④根据终端用户的需要，设计和编制各种功能强劲的应用程序。

(3) 数据库管理员

数据库管理员(Database Administrator, DBA)指全面负责数据库系统的日常管理、维护和运行的人员。DBA 处于终端用户与应用程序员之间，是数据库系统能否正常运转的关键，大型数据库系统需要设置专门的管理办公室。其职责是监督控制数据库的使用和运行，实施数据库系统的维护、改进和重组，开展信息社会化服务。

对于不同规模的数据库系统，用户的人员配置是不相同的。只有大型数据库系统才配备有应用程序员和数据库管理员。应用型微机数据库系统比较简单，其用户通常兼有终端用户和数据库管理员的职能，但必要时也应当兼有应用程序员的能力。

5. 数据库系统

数据库系统是指在计算机系统中引入数据库后构成的系统，一般由数据库、数据库管理系统(及其开发工具)、应用系统、数据库管理员和用户构成。

数据库系统是一个由硬件、软件(操作系统、数据库管理系统和编译系统等)，数据库和用户构成的完整计算机应用系统。数据库是数据库系统的核心和管理对象。因此，数据库系统的含义已经不仅仅是一个对数据进行管理的软件，也不仅仅是一个数据库，数据库系统是一个实际运行的，按照数据库方式存储、维护和向应用系统提供数据支持的系统，数据库系统软硬件的组成如图 1-1 所示。

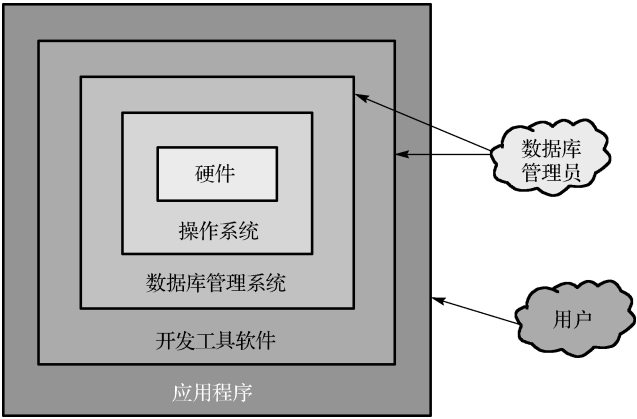


图 1-1 数据库系统软、硬件的组成

(1) 硬件

数据库系统的硬件包括计算机的主机、键盘、显示器和外围设备(如打印机、光盘机、磁带机等)。由于一般数据库系统所存放和处理的数据量很大，加之 DBMS 丰富的功能软件，使得自身所占用的存储空间很大，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：①有足够大的内存以存放操作系统、DBMS 的核心模块、数据缓冲区和应用程序；②有足够大的直接存取设备存放数据(如磁盘)，有足够的存储设备来进行数据备份；③要求计算机有较高的数据传输能力，以提高数据传送率。

(2) 软件

数据库系统的软件除了数据库管理系统之外，还包括操作系统各种高级语言处理程序(编译或解释程序)、应用开发工具软件和特定应用软件等。应用开发工具包括应用程序生成器和第四代语言等高效率、多功能的软件工具，如报表生成系统、表格软件、图形编辑系统等。它们为数据库系统的应用开发人员和最终用户提供了有力的支持。特定应用软件是指为特定用户开发的数据库应用软件，如基于数据库的各种管理软件、管理信息系统(MIS)、决策支持系统(DSS)和办公自动化(OA)等。

1.1.2 数据库管理技术的发展

随着计算机技术的发展，特别是在计算机软件、硬件与网络技术发展的前提下，人们的数据处理要求不断提高，在此情况下，数据管理技术也不断改进。数据库技术是计算机科学技术中发展最快的领域之一，也是应用最广的技术之一，它成为计算机信息系统与应用系统的核心技术和重要基础。数据管理的水平是和计算机硬件、软件的发展相适应的，随着计算机技术的发展，人们的数据管理技术经历了三个阶段的发展：人工管理阶段、文件系统阶段、数据库系统阶段。

1. 人工管理阶段

20 世纪 50 年代中期以前，计算机主要用于科学计算。硬件方面，计算机的外存只有磁带、卡片、纸带，没有磁盘等直接存取的存储设备，存储量非常小；软件方面，没有操作系统，没有高级语言，数据处理的方式是批处理，即机器一次处理一批数据，直到运算完成为止，然后才能进行另外一批数据的处理，中间不能被打断，原因是此时的外存(如磁带、卡片等)只能顺序输入。

人工管理阶段数据管理特点：数据不保存，没有对数据进行管理的软件系统，没有文件的概念，数据不具有独立性。所以有人也称这一数据管理阶段为无管理阶段。

2. 文件系统阶段

这一阶段的主要标志是计算机中有了专门管理数据库的软件——操作系统(文件管理)。

20 世纪 50 年代后期到 60 年代中期，数据管理发展到文件系统阶段。此时的计算机不仅用于科学计算，还大量用于管理。外存储器有了磁盘等直接存取的存储设备。在软件方面，操作系统中已有了专门的管理数据软件，称为文件系统。从处理方式上讲，不仅有了文件批处理，而且能够联机实时处理，联机实时处理是指在需要的时候随时从存储设备中查询、修改或更新，因为操作系统的文件管理功能提供了这种可能。

但由于数据的组织仍然是面向程序的，所以存在大量的数据冗余；而且数据的逻辑结构不能方便地修改和扩充，数据逻辑结构的每一点微小改变都会影响到应用程序。由于文件之间互相独立，因而它们不能反映现实世界中事物之间的联系，操作系统不负责维护文件之间的联系信息。如果文件之间有内容上的联系，那也只能由应用程序去处理。

文件系统阶段数据管理特点：数据可以长期保存，由文件系统管理数据，文件的形式已经多样化，数据具有一定的独立性。

3. 数据库系统阶段

从 20 世纪 60 年代后期开始，数据管理进入数据库系统阶段。这一时期用计算机管理的规模日益庞大，应用越来越广泛，数据量急剧增长，数据要求共享的呼声越来越大。这种共享的含义是多种应用、多种语言互相覆盖地共享数据集合。此时的计算机有了大容量磁盘，计算能力也非常强。硬件价格下降，编制软件和维护软件的费用相对在增加。联机实时处理的要求更多，并开始提出和考虑并行处理。

在这样的背景下，数据管理技术进入数据库系统阶段。

现实世界是复杂的，反映现实世界的各类数据之间必然存在错综复杂的联系。为反映这种复杂的数据结构，让数据资源能为多种应用需要服务，并为多个用户所共享，同时为让用户能更方便地使用这些数据资源，在计算机科学中，逐渐形成了数据库技术这一独立分支。计算机中的数据及数据的管理统一由数据库系统来完成。

数据库系统的目标是解决数据冗余问题，实现数据独立性，实现数据共享并解决由于数据共享而带来的数据完整性、安全性及并发控制等一系列问题。为实现这一目标，数据库的运行必须有一个软件系统来控制，这个系统软件称为数据库管理系统(Database Management System, DBMS)。数据库管理系统将程序员进一步解脱出来，就像当初操作系统将程序员从直接控制物理读写中解脱出来一样。程序员此时不需要再考虑数据中的数据是不是因为改动而造成不一致，也不用担心由于应用功能的扩充，而导致程序重写，数据结构重新变动。

在这一阶段，数据管理具有下面的优点：

数据结构化是数据库系统与文件系统的根本区别。采用数据模型表示复杂的数据结构，数据模型不仅描述数据本身的特征，还要描述数据之间的联系，这种联系通过存取路径来实现。这样，数据不再面向特定的某个或多个应用，而是面向整个应用系统。数据冗余明显减少，实现了数据共享。

(1) 数据共享性高

数据库从整体的观点来看待和描述数据，数据不再是面向某一应用，而是面向整个系统。

这样就减小了数据的冗余,节约存储空间,缩短存取时间,避免数据之间的不相容和不一致。对数据库的应用可以很灵活,面向不同的应用,存取相应的数据库的子集。当应用需求改变或增加时,只要重新选择数据子集或者加上一部分数据,便可以满足更多更新的要求,也就是保证了系统的易扩充性。

(2) 数据独立性高

数据库提供数据的存储结构与逻辑结构之间的映像或转换功能,使得当数据的物理存储结构改变时,数据的逻辑结构可以不变,从而程序也不用改变。这就是数据与程序的物理独立性。也就是说,程序面向逻辑数据结构,不去考虑物理的数据存放形式。数据库可以保证数据的物理改变不引起逻辑结构的改变。

数据库还提供了数据的总体逻辑结构与某类应用所涉及的局部逻辑结构之间的映像或转换功能。当总体的逻辑结构改变时,局部逻辑结构可以通过这种映像的转换保持不变,从而程序也不用改变。这就是数据与程序的逻辑独立性。

(3) 统一的数据管理和控制功能,包括数据的安全性控制、数据的完整性控制及并发控制、数据库恢复。

数据库是多用户共享的数据资源。对数据库的使用经常是并发的。为保证数据的安全可靠和正确有效,数据库管理系统必须提供一定的功能来保证。

数据库的安全性是指防治非法用户的非法使用数据库而提供的保护。比如,不是学校的成员不允许使用教务管理系统,学生允许读取成绩但不允许修改成绩等。

数据的完整性是指数据的正确性和兼容性。数据库管理系统必须保证数据库的数据满足规定的约束条件,常见的有对数据值的约束条件。比如在建立上面的例子中的数据库时,数据库管理系统必须保证输入的成绩值大于 0,否则,系统发出警告。

数据的并发控制是多用户共享数据库必须解决的问题。要说明并发操作对数据的影响,必须首先明确,数据库是保存在外存中的数据资源,而用户对数据库的操作是先读入内存操作,修改数据时,是在内存修改读入的数据复本,然后再将这个复本写回到存储的数据库中,实现物理的改变。

4. 分布式数据库系统

分布式数据库系统是一个逻辑上统一、地域上分散的数据集合,是计算机网络环境中各个局部数据库的逻辑集合,同时受分布式数据库管理系统的控制和管理。

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。分布式数据库系统适合于单位分散的部门,允许各个部门将其常用的数据存储在本地,实施就地存放本地使用,从而提高响应速度,如银行业务、飞机订票、火车订票等,分布式数据库具有以下几个特点。

- 数据独立性与位置透明性
- 集中和节点自治相结合
- 一致性和可恢复性
- 复制透明性
- 易于扩展性

5. 并行数据库系统

并行数据库系统(Parallel Database System)是新一代高性能的数据库系统,从 90 年代至

今,随着处理器、存储、网络等相关基础技术的发展,并行数据库技术的研究重点在数据操作的时间并行性和空间并行性上。并行数据库系统具有如下特点:

- 高性能
- 高可用性
- 可扩充性

数据库是现代信息系统不可分割的重要组成部分,大量数据库已经普遍存在于科学技术、工业、农业、商业、服务业和政府部门的信息系统中。数据库技术的发展是沿着数据模型的主线展开的。

1.2 数据模型

1.2.1 数据模型的基本概念

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头脑中来,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,又与特定的 DBMS 无关的概念模型,然后再把概念模型转换为计算机上某一 DBMS 支持的数据模型。

数据模型所描述的内容包括三个部分:数据结构、数据操作、数据约束。

(1) 数据结构:数据模型中的数据结构主要描述数据的类型、内容、性质以及数据间的联系等。数据结构是数据模型的基础,数据操作和约束都基本建立在数据结构上。不同的数据结构具有不同的操作和约束。

(2) 数据操作:数据模型中的数据操作主要描述在相应的数据结构上的操作类型和操作方式。

(3) 数据约束:数据模型中的数据约束主要描述数据结构内数据间的语法、词义联系、它们之间的制约和依存关系,以及数据动态变化的规则,以保证数据的正确、有效和相容。

数据模型按不同的应用层次分成三种类型,分别是概念数据模型 (Conceptual Data Model)、逻辑数据模型 (Logical Data Model)、物理数据模型 (Physical Data Model)。

(1) 概念模型:是面向数据库用户的现实世界的模型,主要用来描述世界的概念化结构,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,与具体的数据管理系统无关。概念数据模型必须换成逻辑数据模型,才能在 DBMS 中实现。

在概念数据模型中最常用的是 E-R 模型、面向对象模型及谓词模型等。

(2) 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,如网状模型、层次模型、关系模型等。此模型既要面向用户,又要面向系统,主要用于数据库管理系统 (DBMS) 的实现。

(3) 物理模型:是面向计算机物理表示的模型,描述了数据在存储介质上的组织结构,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其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由系统自动完成,而设计者只设计索引、聚集等特殊结构。

1.2.2 信息世界中的基本概念

概念数据模型用来建立信息世界的数据库模型，强调语义表达，描述信息结构，是对现实世界的第一层抽象。信息世界的一些基本概念包括实体、属性、域、实体型、实体集、联系等。

1. 实体

实体是具有相同属性描述的对象(人、地点、事物)的集合。实体是现实世界中客观存在的、能相互区别的任何事物，实体可以是实际的事物，也可以是实际的事件。例如，学生、教师、课本等是实际事物，而授课、借阅图书等则是实际的事件。凡是有共性的实体可组成一个集合，称为实体集。例如，若干个学生实体的集合构成学生实体集。

2. 属性与域

属性就是对一个对象的抽象刻画。一个具体事物，总是有许许多多的性质与关系，我们把一个事物的性质与关系都叫做事物的属性。属性刻画了实体的特征。一个实体可以有若干个属性，实体及其所有的属性构成了实体的一个完整描述。因此实体与属性间有一定的关联。例如，在学生档案中每个学生(实体)可以有学号、姓名、性别、年龄等若干属性，它们组成了一个有关学生(实体)的完整描述。

一个实体往往可以有若干个属性。每个属性可以有值，如梁西川出生日期取值为“1990-12-1”，史晓庆的政治面貌是“中共党员”，一个属性的取值范围称为该属性的值域。在学生表(见表 1-1)中，每一行表示一个实体，这个实体可以用一组属性值表示。比如，(20100102, 梁西川, 男, 1990-12-1, 群众, 201001)表示一个学生实体。

表 1-1 学生表

学 号	姓 名	性 别	出 生 日 期	政 治 面 貌	班 级 编 号
20100102	梁西川	男	1990-12-1	群众	201001
20100104	史晓庆	女	1991-4-3	中共党员	201001
20100105	王昭宇	男	1991-11-21	群众	201103
20100201	余婷婷	女	1991-10-3	群众	201002

3. 实体型与实体集

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，出生日期，政治面貌，班级编号)就是一个实体型。

同型实体的集合称为实体集。例如，全体学生就是一个实体集。

4. 实体之间的联系

两个实体集之间实体的对应关系称为联系，实体之间的联系可归结为如下三种。

(1)一对一联系，如图 1-2 所示，如果实体集 E1 中的每一个实体至多和实体集 E2 中的一个实体有联系，反之亦然，则称 E1 和 E2 是一一对一的联系，表示为 1：1。例如，实体集校长和实体集学校之间的联系是一对一。

(2)一对多联系，如图 1-3 所示，如果实体集 E1 中的每个实体与实体集 E2 中的任意个实体有联系，而实体集 E2 中的每一个实体至多和实体集 E1 中的一个实体有联系，则称 E1 和 E2 之间是一对多的联系，表示为 1：n，E1 称为一方，E2 为多方。

例如，实体集学校和实体集学生之间是一对多的联系，一方是实体集学校，多方是实体集学生，即一个学校对应多个学生。

(3) 多对多联系，如图 1-4 所示，如果实体集 E1 中的每个实体与实体集 E2 中的任意个实体有联系，反之，实体集 E2 中的每个实体与实体集 E1 中的任意个实体有联系，则称 E1 和 E2 之间是多对多的联系，表示为 $m:n$ 。

例如，实体集学生和实体集课程之间是多对多的关系，即一个学生可以学多门课程，而每门课程可以由多个学生来学。联系也可能有属性。例如，学生“学”某门课程所取得的成绩，既不是学生的属性也不是课程的属性。由于“成绩”既依赖于某名特定的学生又依赖于某门特定的课程，所以它是学生与课程之间的联系“学”的属性。

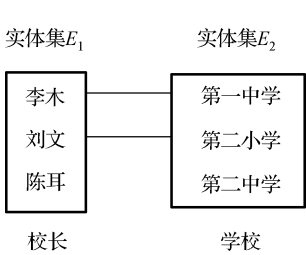


图 1-2 一对一联系示例

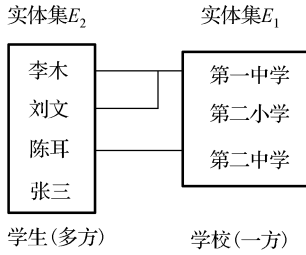


图 1-3 一对多联系示例

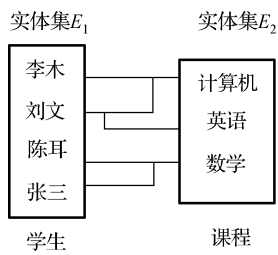


图 1-4 多对多联系示例

1.2.3 E-R 模型

概念模型是对信息世界建模，所以概念模型应该能够方便、准确地表示出上述信息世界中的常用概念。概念模型的表示方法很多，其中尤为著名的是 Peter Chen 提出的“实体-联系方法” (Entity-Relationship Approach)，简称 E-R 模型。该方法用图形方式表示实体集之间的联系。

E-R 图的基本图素包括实体、属性、联系和连线四种基本图素，如图 1-5 所示。

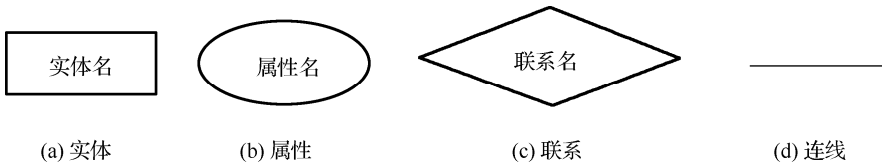


图 1-5 E-R 基本图素

- (1) 矩形：表示实体集，实体名称写在框内。
- (2) 椭圆：表示实体集或联系的属性，框内标明属性的名称。
- (3) 菱形：表示实体间的关系，框内注明联系名称。
- (4) 连线：连接实体和各个属性、实体和联系，并注明联系种类，即 $1:1$ ，如图 1-6 所示； $1:n$ ，如图 1-7 所示； $n:m$ ，如图 1-8 所示。



图 1-6 两实体 1:1 联系



图 1-7 两实体 1：n 联系



图 1-8 两实体 n：m 联系

【例 1-1】 学生实体具有学号、姓名、性别、出生日期、政治面貌和班级编号属性，用 E-R 图元素表示学生实体及属性。

学生实体及属性如图 1-9 所示。

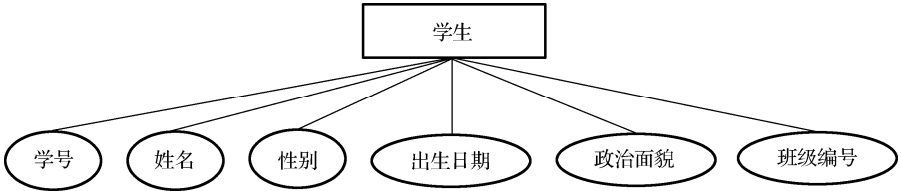


图 1-9 学生实体及属性

需要注意的是，一个联系若具有属性，则这些属性也要用无向边与该联系连接起来，如例 1-2 所示。

【例 1-2】 用 E-R 图表示学生成绩管理系统中学生与课程的联系。绘图步骤如下。

(1) 分析：每个学生选修若干课程，每门课可由若干学生选修，联系为多对多，学生和课程的联系是“选修”，联系“选修”产生属性“分数”。

(2) 确定实体型及其属性。学生成绩管理系统中包含学生与课程两个实体型。其中学生的属性有学号、姓名、性别、出生日期、政治面貌和班级编号。课程的属性有课程编号、课程名称、课程类别、学分。分别用矩形和椭圆表示两个实体型及其属性。

(3) 确定这两个实体型之间的联系为“选修”。选修后产生分数，分别用菱形和椭圆表示联系及其属性。

(4) 对实体型和联系用连线组合，并标上联系方式 $n:m$ ，得到学生选修课程的 E-R 图，如图 1-10 所示。

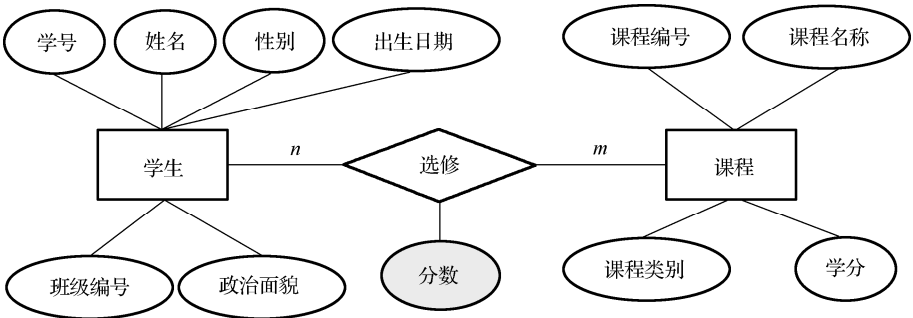


图 1-10 学生选修课程的 E-R 图

【例 1-3】 用 E-R 图表示网上购物系统中顾客与商品的联系。绘图步骤如下。

(1) 网上购物系统中有顾客和商品两个实体型。其中顾客的属性有顾客 ID、密码、姓名和账号余额。商品的属性有商品 ID、商品名称、单价和库存量。分别用矩形和椭圆形表示两个实体型及其属性。

(2) 这两个实体型之间的联系为“购买”。购买后产生数量、时间和送货方式，分别用菱形和椭圆表示联系及其属性。

(3) 一个顾客可购买多个商品，一件商品也可以被多个顾客购买，两个实体型的联系为多对多，标上联系类型 $n:m$ ，得到网上购物系统中顾客购买商品的 E-R 图，如图 1-11 所示。

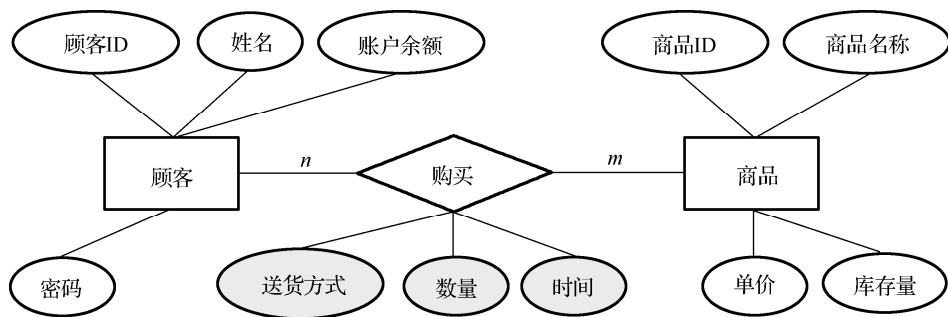


图 1-11 顾客与商品的 E-R 图

单个或更多实体型之间也有类似于两个实体型之间的三种联系类型。

如对于学生、课程和教师这三个实体，若规定每个学生可以选择多门课程，每门课程可以被多个学生选修，每门课程唯一对应一个任课教师，一个教师可以讲授多门课程。则学生、课程和教师这三个实体之间的联系如图 1-12 所示。

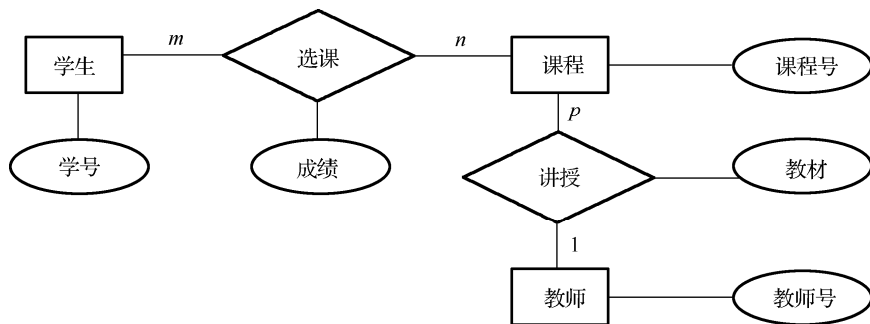


图 1-12 三个实体型关系示例

又如，有三个实体：项目、零件和厂家，假设厂家供应零件，每种零件可由多个厂家供应给不同的工程项目，仓库负责采购零件并管理零件的入库、出库，多个工程项目所需的零件在仓库领取。则这三个实体型间是多对多联系，如图 1-13 所示。



图 1-13 多个不同实体型间联系示例

同一个实体型对应的实体集内容也存在三种对应关系，可以把一个实体集逻辑上看成两

个与原来一样的实体集来理解。例如，职工实体集内部就有领导和被领导的联系，即一职工领导若干名职工，而一个职工仅被另外一名职工直接领导，是一对多联系，如图 1-14 所示。

1.2.4 层次模型

层次模型(Hierarchical Model)是数据库系统中最早采用的数据模型，它是通过从属关系结构表示数据间的联系，层次模型是有向“树”结构。用树形表示数据之间的多级层次结构。其结构特点：

- (1) 只有一个最高结点即根结点；
- (2) 其余结点有且仅有一个父结点；
- (3) 上下层结点之间表示一对多的联系。

例如，行政组织机构(见图 1-15)、家族辈分关系等都是典型的层次模型。

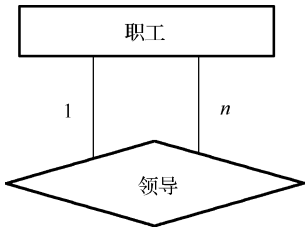


图 1-14 单个实体型内部 1:n 联系

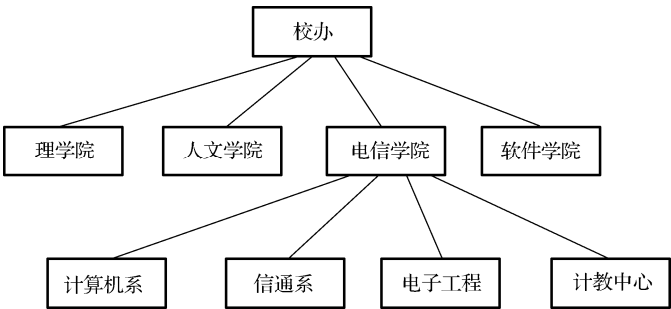


图 1-15 层次模型示例

1.2.5 网状模型

网状模型(Network Model)是层次模型的扩展，是一种更具有普遍性的结构，它表示多个从属关系的层次结构，呈现一种交叉关系的网络结构，网状模型是有向“图”结构。

它满足以下条件：

- (1) 用图表示数据之间的关系；
- (2) 允许结点有多于一个的父结点；
- (3) 可以有一个以上的结点没有父结点。

如图 1-16 所示的网状模型示例，出版社和科研机构没有父节点，有多个根节点，图书有多个父节点。

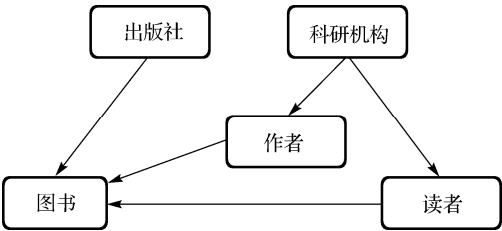


图 1-16 网状模型示例

1.2.6 关系模型

关系模型(Relational Model)是用一组二维表来表示数据和数据之间的联系的数据模型。每一张二维表组成一个关系，一个关系有一个关系名。一个关系由表头和记录两部分组成，表头由描述客观世界的各个属性组成，每条记录的数据由实体在各个字段的值组成。表 1-2 所示为班级关系。

表 1-2 班级关系

班 级 编 号	班 级 名 称	入 学 时 间	专 业	培 养 层 次
201001	2010 级会计学 1 班	2010-9-1	会计学	本科
201002	2010 级国际经济与贸易 1 班	2010-9-1	国际经济与贸易	本科
201101	2011 级会计学 1 班	2011-9-1	会计学	本科
201102	2011 级工商管理 1 班	2011-9-1	工商管理	本科

在关系模型建立的数据库中实体和各类联系都用关系来表示，对数据的检索结果也是关系，概念单一，从而使得数据库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。

关系结构简单、直观，在数据库技术中，将支持关系模型的数据库管理系统称为关系型数据库，目前关系型数据库在数据库管理领域占主导地位。

1.3 关系数据库

1.3.1 关系性质与特点

1. 基本概念

(1) 关系

一个关系(Relation)就是一个二维表。每个关系都有一个关系名。对关系的结构描述称为关系模式，其格式为：关系名(属性名 1，属性名 2，…，属性名 *n*)。例如，

学生(学号，姓名，性别，出生日期，政治面貌，班级编号)

成绩(学号，课程编号，分数)

(2) 元组(Tuple)

在二维表中，从第二行起的每一行称为一个元组，在文件中对应一条具体记录。例如，学生表中的一行对应一条学生记录。

(3) 属性(Attribute)

二维表中，每一列称为一个属性，在文件中对应一个字段。每一列有一个属性名，二维表第一行显示的每一列的名称，在文件中对应字段名，如“名”“性别”等，属性名即字段名。行和列的交叉位置表示某个属性的值。

(4) 域(Domain)

属性的取值范围。如学生表中“性别”字段的取值只能是“男”或“女”，成绩表中的“分数”字段取值范围是 0~100。

(5) 主键(Primary Key)

表中的某个属性或某些属性的集合,能唯一确定一个元组。例如,在学生表中“学号”字段可以唯一标示一条学生记录。该字段就是学生表的主键。在成绩表中,只有属性的组合“学号+课程编号”才能唯一区分每一条记录,则属性集“学号+课程编号”是成绩表的主键。

如果一个二维表中存在多个关键字或码,它们称为该表的候选关键字或候选码。在候选码中指定一个关键字作为用户使用的关键字,称为主关键字或主码。

(6) 外键(Foreign Key)

外键是一个表中的一个属性或属性组,它们在其他表中作为主键而存在。一个表中的外键被认为是对另外一个表中主键的引用。如关系

学生(学号, 姓名, 性别, 出生日期, 政治面貌, 班级编号)

班级(班级编号, 班级名称, 入学时间, 专业, 培养层次)

在班级表中“班级编号”是主键,在学生表中“班级编号”则是外键。

2. 关系的特点

在关系数据库中,对关系有一定的要求和限制,即关系必须符合以下特点。

(1) 关系中的每个属性都必须是不可分解的,是最基本的数据单元,即数据表中不能再包含表。

(2) 一个关系中不允许有相同的属性名,即在定义表结构时,一张表中不能出现重复的字段名,但允许不同的表中有同名属性。

(3) 关系中不允许出现相同的元组,即数据表中任意两行不能完全相同。否则不仅会增加数据量,造成数据的“冗余”,还会造成数据查询和统计的错误,产生数据不一致问题。因此,应绝对避免元组重复现象,确保实体的唯一性和完整性。

(4) 关系中同一列的数据类型必须相同,也就是说,数据表中任一字段的取值范围应属于同一个域。例如,学生选课成绩表中的“成绩”属性值不能既有百分制,又有五分制,必须统一成一种语义。

(5) 关系中行、列的次序任意,任意交换两行或两列的位置并不影响数据的实际含义。

3. E-R 模型向关系模型的转换

要建立数据库,就需要 E-R 模型向关系模型进行转换,在 E-R 模型中的一个实体将转换为一个关系。实体的属性就是关系的属性,实体的标识符就是关系的字段名。在关系模型中,可以通过各关系中具有相同意义的属性来建立实体之间的联系。

那么,E-R 图转化成关系模式的步骤是什么呢?

前提,是已经把需求中的实体,以及实体中的联系确定。

第一,把每个实体都转化成关系模式 $R(A、B)$ 形式($A、B$ 代表属性)。

第二,实体中的属性即关系模式中的属性要求是不可再分的。

第三,也是最重要的步骤,实体之间联系的转换。

实体之间的联系分为 $1:1$, $1:n$, $m:n$ 三种形式。

(1) 实体间 $1:1$ 联系的转换

将两个实体转化成关系模式之后,把任意实体的主键和联系的属性放到另一个实体的关系模式中。

如图 1-17 所示,实体“部门”和实体“经理”之间存在 $1:1$ 联系,要转换成关系模型,

我们可以将实体“部门”转换成关系部门表(部门号, 部门名, 经理号), 实体“经理”转换成关系经理表(经理号, 经理名, 电话), 通过在部门表中增加“经理号”这一属性, 建立两个关系的公共属性, 从而确定两个关系的1:1联系。同理, 也可以转换成部门表(部门号, 部门名)、经理表(经理号, 部门号, 经理名, 电话)。不需要单独将联系“管理”转换成一个关系。

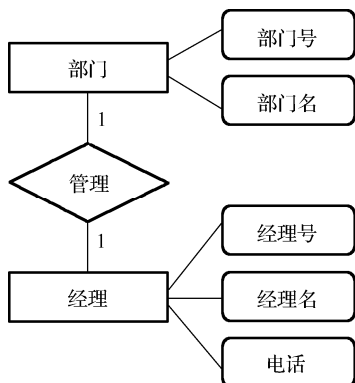


图 1-17 1:1 转换示例

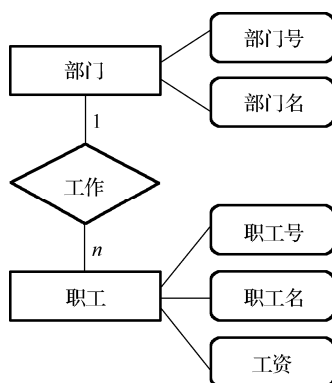


图 1-18 1:n 转换示例

(2) 实体间 1:n 关系的转换

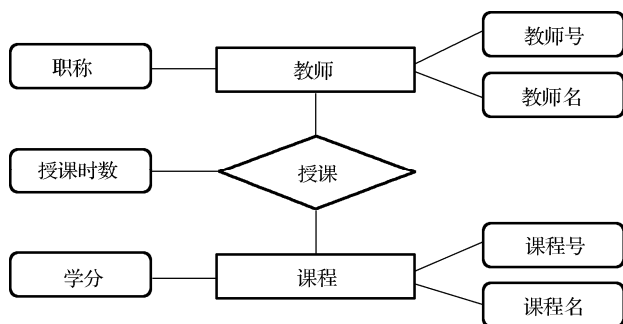
将两个实体各自转化成关系模式后, 把联系数量为 1 的实体的主键和联系的属性放到联系数量为 n 的实体关系模式中。

如图 1-18 所示, 实体“部门”和实体“职工”之间存在 1:n 的联系, 转换成关系模型, 可以转换成部门表(部门号, 部门名)、职工表(职工号, 部门号, 职工名, 工资)。联系“工作”不需要单独转换成关系表, 只需要在职工表中增加“部门号”这一公共属性, 即可实现两个关系之间的一对多联系。

(3) 实体间 $n:m$ 关系的转换

将两个实体各自转换成关系模式后, 把两个实体中的主键和联系的属性放到另一个关系模式中(注意, 多生成一个关系模式)。

如图 1-19 所示, 实体“教师”和实体“课程”之间存在 $n:m$ 的联系, 要转换成关系模型, 可以转换成教师表(教师号, 教师名, 职称)、课程表(课程号, 课程名, 学分)和授课表(教师号, 课程号, 授课时数)。联系“授课”需要单独转换成一个关系(即授课表)才能实现这种多对多联系。

图 1-19 $m:n$ 转换示例

综上所述，在进行转换时，实体间 1：1 的联系和 1：n 的联系一般不转换成一个关系，而一个 m：n 的联系需要转换为一个关系。

4. 关系操作

关系模型的数据操作一般有 4 种：查询、插入、删除及修改。关系模型给出了关系操作的能力，但不 对RDBMS语言给出具体的语法要求。

(1)数据插入(Insert)。数据插入仅对一个关系而言,在指定的关系中插入一个或多个元组。

(2)数据删除(Delete)。数据删除的基本单位是一个表中的元组，它将满足条件的元组从表中删除。数据删除可以分解成一个表内元组选择与表中的元组删除两个基本操作。

(3)数据修改(Update)。数据修改是在一个关系中修改指定的元组与属性。

(4)数据查询(Query)。数据查询是数据操作的核心操作，包括单表查询和多表查询。查询包括选择(select)、投影(project)、连接(join)、除(Divide)、并(Union)、交(Intersection)、差(Difference)等操作。查询的表达能力是其中最重要的部分。

关系操作的特点是集合操作方式，即操作的对象和结构都是集合。

1.3.2 关系运算

关系代数是 以关系为运算对象的一组高级运算的集合。关系代数的运算对象和结果都是关系，关系代数的运算符包括 4 类：集合运算符、专门关系运算符、算术比较符、逻辑运算符，如表 1-3 所示。

表 1-3 关系代数运算符

运 算 符		含 义	运 算 符		含 义
集合运算符	∪	并	比较运算符	>	大于
	-	差		≥	大于等于
	∩	交		<	小于
	×	广义笛卡儿积		≤	小于等于
专门的关系运算符	σ	选择	逻辑运算符	=	等于
	π	投影		≠	不等于
		联结		¬	非
	÷	除		∧	与
				∨	或

关系定义为元数相同的元组的集合。集合中的元素为元组，关系代数中的操作可分为传统的集合运算和专门的关系运算两类。

1. 传统的集合运算

传统的集合运算包括并、差、交、笛卡儿积 4 种运算[运算从关系的水平(行)的角度来进行]。

(1)并

设两个关系 R 和 S 具有相同的 关系模式,R 和 S 的并是由属于 R 和 S 的元组构成的集合,记为 R∪S。

注意：R 和 S 的元数相同，即两个关系的结构相同。

例如，关系 R：

商 品 代 码	子公司代码	品 名	数 量	单 价
1	Comp1	钢笔	50	10.00
2	Comp1	圆珠笔	200	6.00

关系 S:

商 品 代 码	子公司代码	品 名	数 量	单 价
1	Comp1	钢笔	50	10.00
5	Comp2	练习本	200	3.00
6	Comp2	信箋	1000	3.00

RUS 结果:

商 品 代 码	子公司代码	品 名	数 量	单 价
1	Comp1	钢笔	50	10.00
2	Comp1	圆珠笔	200	6.00
5	Comp2	练习本	200	3.00
6	Comp2	信箋	1000	3.00

(2) 交

设两个关系 R 和 S 具有相同的模式，R 和 S 的交是由同时属于 R 和 S 的元组构成的集合，记为 $R \cap S$ 。

例如，关系 R:

学 号	课 程 名	分 数
1	数学	80
1	英语	85
1	政治	90
2	数学	85
2	英语	80
2	政治	90

关系 S:

学 号	课 名	分 数
1	数学	80
1	英语	85
1	历史	92
2	数学	85
2	英语	80
2	历史	80

$R \cap S$ 结果:

学 号	课 名	分 数
1	数学	80
1	英语	85
2	数学	85
2	英语	80

(3) 差

设两个关系 R 和 S 具有相同的模式，R 和 S 的差是由属于 R 但不属于 S 的元组构成的集合，记为 R-S。

如上例中，R-S 结果：

学 号	课 名	分 数
1	政治	90
2	政治	90

(4) 笛卡尔积

设关系 R 和 S 的元数分别为 r 和 s 。定义 R 和 S 的笛卡儿积是个 $(r+s)$ 元的元组集合，每个元组的前 r 个分量(属性值)来自 R 的一个元组，后 s 个分量来自 S 的一个元组，记为 $R \times S$ 。若 R 有 m 个元组，S 有 n 个元组，则 $R \times S$ 有 $m \times n$ 个元组。

数据示例如下。

关系 R：

姓 名	年 龄
张三	20
李四	19

关系 S：

地 址	邮 编
长沙	410004
娄底	417119

$R \times S$ 结果：

姓 名	年 龄	地 址	邮 编
张三	20	长沙	410004
张三	20	娄底	417119
李四	19	长沙	410004
李四	19	娄底	417119

2. 专门的关系运算

专门的关系运算包括选择、投影、连接、除等。

(1) 选择

从关系中找出满足给定条件的所有元组称为选择。其中的条件是以逻辑表达式给出的，该逻辑表达式的值为真的元组被选取。这是从行的角度进行的运算，即水平方向抽取元组。经过选择运算得到的结果能形成新的关系，其关系模式不变，但其中元组的数目小于或等于原来的关系中元组的个数，是原关系的一个子集。

例如，在关系 R 中选择分数在 90 分以上的记录。

学 号	课 程 名	分 数
1	数学	80
1	英语	85
1	政治	90
2	数学	85
2	英语	80
2	政治	90