

第 1 章 数据库系统概论

数据库是数据管理的最新技术，是计算机学科的重要分支。十余年来，数据库管理系统已从专业的应用程序包发展成为通用的系统软件。由于数据库系统具有数据结构化、最低冗余度、较高的程序与数据独立性等优点，较大的信息管理系统都是以数据库作为基础的。数据管理技术的发展经历了 3 个阶段：

- 人工管理阶段
- 文件系统阶段
- 数据库系统阶段

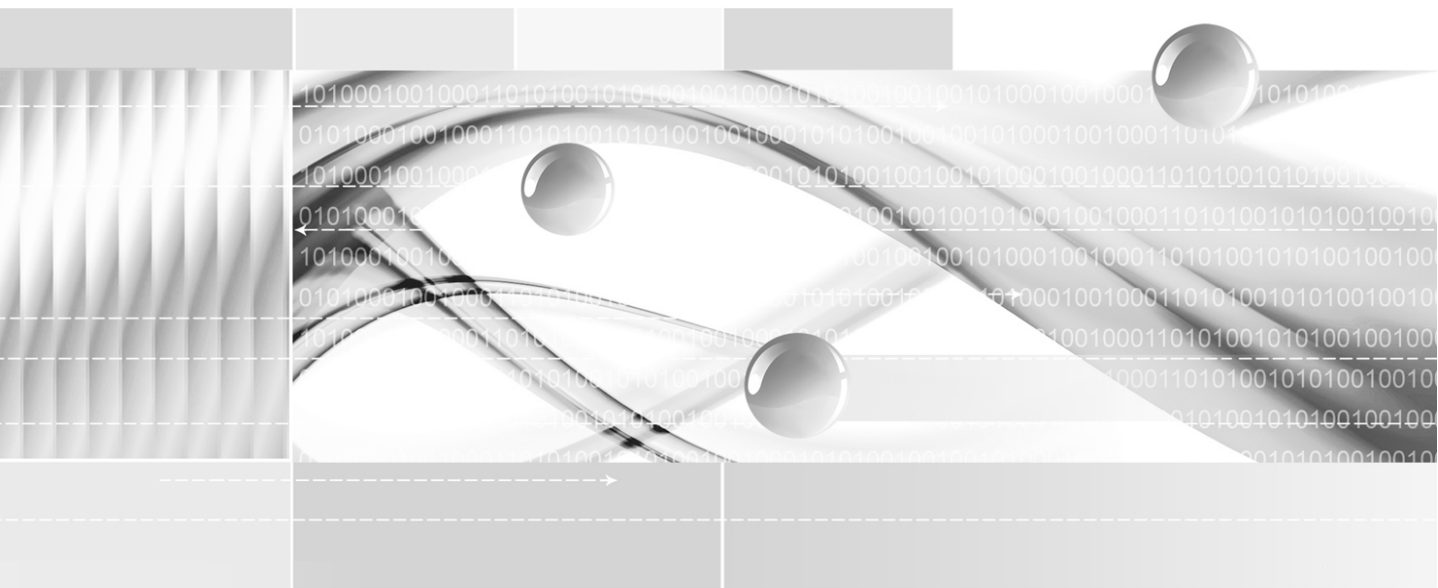
数据库作为数据管理技术发展的最新阶段，其体系结构由 3 级模式构成：

- 模式
- 外模式
- 内模式

在这一章里，我们将学习数据库系统的基本概念和术语，了解数据管理技术的发展历史，学习数据库系统的体系结构，包括三级模式与两级映像，了解数据库的语言，为后续的学习打下一个良好的基础。

学习目标：

- 掌握数据库的基本概念和相关术语
- 了解数据库技术的产生与发展
- 掌握数据库管理技术发展的三个阶段
- 理解数据库中的各种数据模型
- 了解数据库系统的体系结构和一般组成
- 理解数据库系统的模式结构
- 了解数据库的各种语言



数据库和数据库系统已经成为现代社会日常生活中的重要组成部分,在日常的工作和生活中,大多数人都会或多或少地与数据库打交道。比如,到银行存钱或取钱;预定机票或宾馆房间;在图书馆的计算机管理系统中查找图书条目;或者从网上商店购买商品,所有这些活动都会涉及访问数据库系统。

在传统的数据库应用中,大多数信息都是以文本或数字形式来存储和访问的。随着技术的发展,不断出现许多新的数据库系统应用。多媒体数据库可以存储图片、视屏片段以及语音消息。地理信息系统可以存储和分析地图、气象数据和卫星图像。许多公司和政府机构使用数据仓库和联机分析处理来提取、分析大型数据库中的有用信息以辅助决策。实时数据库和主动数据库技术则用于控制工业和制造业的生产过程。数据库搜索技术还用到万维网上,改善信息搜索,满足互联网用户查找信息的要求。在本章,我们将介绍数据库应用的基础知识,为后续的学习打下基础。



1.1 数据库系统的应用及其研究领域

数据库的应用领域非常广泛,不管是学校、公司或大型企业,还是政府部门,都需要使用数据库来存储数据信息。例如,学校要存储学生的基本信息、课程信息和成绩等;医院可以用数据库来存储医生、病人、药品的基本信息,以及病人的就诊信息等;银行需要用数据库来存储客户的信息、账户以及银行的交易记录;电信业需要数据库来存储通话记录,产生每月清单,维护预付电话卡的余额和存储通信网络的信息;金融业需要数据库来存储股票、债券等金融票据的持有、出售和买入的信息……数据库已经渗入到社会日常生活中的各个方面。

传统数据库中的很大一部分用于商务领域,如证券行业、银行、销售部门、医院、公司或企业单位,以及国家政府部门、国防军工领域、科技发展领域等。随着信息时代的发展,数据库也相应产生了一些新的应用领域,主要表现在下面6个方面。

1. 多媒体数据库

这类数据库主要存储与多媒体相关的数据,如声音、图像和视频等数据。多媒体数据最大的特点是数据连续,而且数据量比较大,存储需要的空间较大。

2. 移动数据库

这类数据库是在移动计算机系统中发展起来的,如笔记本电脑、掌上计算机等。该数据库最大的特点是通过无线数字通信网络传输的。移动数据库可以随时随地地获取和访问数据,为一些商务应用和一些紧急情况带来了很大的便利。

3. 空间数据库

这类数据库目前发展比较迅速。它主要包括地理信息数据库(又称为地理信息系统,即GIS)和计算机辅助设计(CAD)数据库。其中地理信息数据库一般存储与地图相关的信息数据;计算机辅助设计数据库一般存储设计信息的空间数据库,如机械、集成电路以及电子设备设计图等。

4. 信息检索系统

信息检索就是根据用户输入的信息,从数据库中查找相关的文档或信息,并把查找的信息反馈给用户。信息检索领域和数据库是同步发展的,它是一种典型的联机文档管理系统或者联机图书目录。

5. 分布式信息检索

这类数据库是随着Internet的发展而产生的数据库。它一般用于因特网及远距离计算机网络系统中。特别是随着电子商务的发展,这类数据库发展更加迅猛。许多网络用户(如个人、公司或企业等)在自己的计算机中存储信息,同时希望通过网络使用发送电子邮件、文件传输、远程登录方式和别人共享这些信息。分布式信息检索满足了这一要求。

6. 专家决策系统

专家决策系统也是数据库应用的一部分。由于越来越多的数据可以联机获取，特别是企业通过这些数据可以对企业的发展做出更好的决策，以使企业更好地运行。由于人工智能的发展，使得专家决策系统的应用更加广泛。



1.2 数据库系统概述



1.2.1 数据库的基本概念和相关术语

在系统地介绍数据库的基本概念之前，这里首先介绍一些数据库最常用的术语和基本概念。

1. 数据、信息与数据处理

在计算机应用中，数据处理和以数据处理为基础的信息系统占据着很大的比重。

人类的一切活动都离不开数据，离不开信息。在不同的领域里，信息的含义有所不同。一般认为信息是数据、消息中包含的意义。数据和信息有时可以混用，例如，数据处理也称为信息处理；有时必须分清，例如，不能把信息系统称为数据系统。

(1) 数据 (Data)

数据是一种符号序列，它的内容是事物特性的反映。数据是对现实世界的事物采用计算机能够识别、存储和处理的方式进行的描述，或者说是计算机化的信息。数据的概念在数据处理领域中得到不断的发展，目前数据不仅包括数字、字母、文字和其他特殊字符，而且还包括图形、图像、声音等多媒体数据。

在计算机中，为了存储和处理现实世界中的事物，就要抽出对这些事物感兴趣的特征组成一个记录来描述。例如，在员工档案中，如果人们最感兴趣的是员工的姓名、性别、年龄、籍贯、所在部门、进入公司的时间和薪水，那么可以这样描述：

(余颖，女，20，四川，销售部，2008，3000)

这里的员工记录就是数据。对于上面这条员工记录，了解其含义的人会得到如下信息：余颖是公司职员，女，20岁，四川人，2008年进入公司，月薪3000元。而不了解其语意的人则无法理解其含义。

(2) 信息

信息是经过加工处理的数据，是人们消化理解了的数据，是数据的具体含义。数据与信息既有联系又有区别。数据是信息的符号表示或载体，而信息则是数据的具体含义，对数据的语义解释。而且同一数据也可能有不同的解释。数据一般都可以表示成某种信息，但并非任何数据都能包含对人们来说有用的信息。信息是抽象的，不随数据设备所决定的数据形式而变化；而数据的表现形式却具有可选择性。

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。例如，同样的新闻可以通过报纸、电台和电视来报道，它的表现形式不同，但其信息的内容可以相同。

(3) 数据处理

数据处理是指将数据转换成信息的过程。广义地讲，它包括对数据的收集、存储、加工、分类、检索、传播等一系列活动。狭义地讲，它是指对所输入的数据进行加工整理。基本目的是从大量、已知数据出发，根据事物之间的固有联系和规律，通过分析归纳、演绎推导等手段，提出对人们有价值、有意义的信息，作为决策的依据。数据的加工可以简单，也可以相当复杂。简单加工包括组织、编码、分类、排序等；复杂加工可以使用统计学方法、数学模型等对数据进行深层次的加工。

因此，可得：信息 = 数据 + 数据处理。



2. 数据库的基本概念

数据库是什么呢?顾名思义,即存放数据的仓库。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。实质上,数据库就是一些存在了很长时间——常常是许多年的信息的聚集。通常意义下,“数据库”这个术语是指由数据库管理系统管理的数据集合。

(1) 数据库

数据库(DB, Database)是指长期存储在计算机内的、有组织的、可共享的大量数据的集合。数据库中的数据按照一定的数据模型组织、描述和存储,具有较小的冗余度、较高的数据独立性和易扩展性,并为各种用户共享。

(2) 数据库管理系统

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。完成这个任务的即是数据库管理系统(DBMS)。

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,主要目标是使数据成为方便各种用户使用的资源,并提高数据的安全性、完整性和可用性。

DBMS一般具有如下功能:

① 数据定义功能,提供数据定义语言DDL(Data Definition Language),对各级数据模式进行精确定义。

② 数据操纵功能,提供数据操纵语言DML(Data Manipulation Language),可以对数据库中的数据进行追加、插入、修改、删除、检索等操作。

③ 数据库运行控制功能,提供数据控制语言DCL(Data Control Language),可以对数据库中的数据进行并发控制、数据的安全性控制、数据的完整性控制。

④ 数据组织、存储和管理功能,DBMS分类组织、存储和管理各种数据,包括数据字典、用户数据、数据的存取路径。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,实现数据之间的联系。

⑤ 数据库的建立和维护功能,主要包括数据库初始数据的输入、转换功能,数据库的转储、恢复功能,数据库的重组功能和性能监视、分析功能等。这些功能通常是由一些实用程序或管理工具完成的。

数据库管理系统是数据库系统的一个重要组成部分。

(3) 数据库系统

数据库系统是由数据库及其管理软件组成的系统,一般由数据库、数据库管理系统、数据库管理员(DBA)、用户和应用程序组成。一个数据库系统应该具有如下特征:

① 允许用户使用一种叫作数据定义语言的专用语言,建立新的数据库和指定它们的模式(schema)。

② 使用户能够用适当的语言查询数据和更新数据。

③ 支持存储大量的数据——G(10^9)字节以上,经过很长一段时间以后仍保证其安全,同时允许对数据库合法的查询和更新。

④ 控制多用户的同时访问,使得一个用户的访问不影响其他用户,保证同时访问不会损害数据。

上面介绍了与数据库相关的一些基本的、抽象的概念,接下来介绍一些具体概念。

(4) 实体与表

在数据库系统中,一个实体就是一个人、一个地方、一个事件,或者一个我们将要为其收集数据的物体。客观存在并可相互区别的事物称为实体。比如,在学校中,学生、教师员工、课程等都是实体。学生可以组成一个实体集。一个实体集就是实体的一个特定集合。每个实体都有某些称为属性的

特征。如学生实体可能包含以下属性：学生学号、姓名、性别、入学时间、专业方向等。每个属性必须恰当地命名，以便让用户能够知道它的内容，比如学生实体，属性姓名可以存储为 STU_NAME，性别可以存储为 STU_SEX。

表是二维结构，它包括行和列。一个表包括一组相关的实体——实体集。因此，术语实体集和表经常互换使用。表 1.1 列出了一个表的特征。

表 1.1 表的特征

1	表是二维结构，它包括行和列
2	每个表行描述实体集中的一个实体
3	每个表列描述一个属性，每一列有一个明确的名字
4	每行与列的交叉描述一个数据值
5	每个表必须有一个属性或者一个属性复合体来唯一标识每一行
6	一列中的所有值必须是同一数据格式
7	每列有一个明确的数值范围
8	行和列的排列顺序对 DBMS 并不重要

(5) 关系列表和关系数据库

数据库是一个自描述的数据元素的集合以及这些元素之间的联系，数据库中表之间的关系是数据元素间的重要联系。

① 关系列表

关系数据库使用表来组织数据元素，每一个表对应于一个应用实体，而每行则代表实体的一个示例。例如，医院管理系统中医生实体对应数据库中的表 Doctor，该表中的每一行则代表不同的医生。医生数据表的属性构成见表 1-2。

表 1.2 医生数据表的属性构成

医生编号	医生姓名	医生性别	医生年龄	所属部门	技术等级	工资
------	------	------	------	------	------	----

联系通过将来自于一个表的行标识符（医生编号）出现在一个表示诊断的行中，从而建立了该诊断与这位医生之间的联系。诊断数据表的属性构成见表 1.3。

表 1.3 诊断数据表的属性构成

诊断编号	患者编号	医生编号	症状描述	诊断描述	就诊时间
------	------	------	------	------	------

这种表之间通过属性进行联系，构成了关系列表，是关系数据库的一个基础。

② 关系数据库

1970 年，IBM 研究室的 Ted Codd 发表了一篇具有很大影响的关于关系数据库模型的论文，数据库系统发生了显著的变化。Codd 提出数据库系统应为用户提供这样一种观点，即数据库系统是用一种称为“关系”的表来组织数据的。而在背后，可能有一个很复杂的数据结构，以保证对各种查询的快速响应。但与以前的数据库系统的用户不同，关系数据库系统的用户并不关心数据的存储结构，而是使查询能用很高级的语言来实现，从而大大提高了数据库开发人员的效率。

下面先对关系做一下简单介绍，增进读者对关系模型的了解，同时，我们给出一个 SQL 的例子，以便读者了解关系模型如何支持高级语言的查询。

【例 1-1】关系就是表。表的各列以属性开始，属性是列的入口。表 1.4 是一个名为 Doctor（医生）的关系，记录的是医生的信息。

表 1.4 医生数据表

医生编号	医生姓名	医生性别	医生年龄	所属部门	技术等级	工资
001	李红	女	30	101	主任医师	3000
002	张兰	女	35	104	主治医师	1800

假设要查询李红的工资，SQL 的查询语句如下：

```
SELECT 工资
FROM    Doctor
WHERE   医生姓名='李红';
```

1.2.2 数据库技术的产生和发展

数据库技术是应数据库管理任务的需要而产生的。数据管理是指对各种数据进行分类、组织、编码、存储、检索和维护,它是数据处理的中心问题。人们借助计算机进行数据处理是从20世纪60年代开始的。

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。

1. 人工管理阶段

在这一阶段(20世纪50年代中期以前),计算机主要用于科学计算。外部存储器只有磁带、卡片和纸带等,还没有磁盘等直接存取存储设备。软件只有汇编语言,尚无数据管理方面的软件,数据处理方式基本是批处理。这个阶段有如下特点:

① 计算机系统不提供对用户数据的管理功能。用户编制程序时,必须全面考虑好相关的数据,包括数据的定义、存储结构以及存取方法等。程序和数据是一个不可分割的整体。数据脱离了程序就无任何存在的价值,数据无独立性。

② 数据不能共享。不同的程序均有各自的数据,这些数据对不同的程序通常是不相同的,不可共享。即使不同的程序使用了相同的一组数据,这些数据也不能共享,程序中仍然需要各自加入这组数据,谁也不能省略。基于这种数据的不可共享性,必然导致程序与程序之间存在大量的重复数据,增加了数据管理的复杂性,浪费了存储空间。

③ 不单独保存数据。基于数据与程序是一个整体,数据只为本程序所使用,数据只有与相应的程序一起保存才有价值,否则就毫无用处。所以,所有程序的数据均不单独保存。

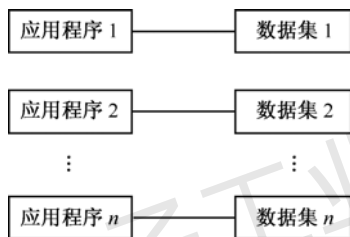


图 1.1 人工管理阶段应用程序与数据之间的对应关系

在人工管理阶段,程序与数据之间的一一对应关系可用图 1.1 表示。

2. 文件系统阶段

在这一阶段(20世纪50年代后期至60年代中期),计算机不仅用于科学计算,还用在信息管理方面。随着数据量的增加,数据的存储、检索和维护问题的解决成为紧迫的需要,数据结构和数据管理技术迅速发展起来。此时,外部存储器已有磁盘、磁鼓等直接存取的存储设备。软件领域出现了操作系统和高级软件。操作系统中的文件系统是专门管理外存的数据管理软件,文件是操作系统管理的重要资源之一。数据处理方式有批处理,也有联机实时处理。这个阶段有如下特点:

① 数据以“文件”形式可长期保存在外部存储器的磁盘上。由于计算机的应用转向信息管理,因此对文件要进行大量的查询、修改和插入等操作。

② 数据的逻辑结构与物理结构有了区别,但比较简单。程序与数据之间具有“设备独立性”,即程序只需用文件名就可与数据打交道,不必关心数据的物理位置。由操作系统的文件系统提供存取方法。文件组织已多样化,有索引文件、链接文件和直接存取文件等,但文件之间相互独立、缺乏联系。数据之间的联系要通过程序去构造。数据不再属于某个特定的程序,可以重复使用,即数据面向应用。文件结构的设计仍然是基于特定的用途。程序基于特定的物理结构和存取方法,因此程序与数据结构之间的依赖关系并未根本改变,对数据的操作以记录为单位。这是由于文件中只存储数据,不存储文件记录的结构描述信息。文件的建立、存取、查询、插入、删除、修改等所有操作,都要用程序来实现。

随着数据管理规模的扩大，数据量急剧增加，文件系统显露出一些缺陷：

① 数据冗余。由于文件之间缺乏联系，造成每个应用程序都有对应的文件，有可能同样的数据在多个文件中重复存储。

② 不一致性。这往往是由数据冗余造成的，在进行更新操作时，稍不谨慎，就可能使同样的数据在不同的文件中不一样。

③ 数据联系弱。这是由于文件之间相互独立、缺乏联系造成的。

文件系统阶段是数据管理技术发展中的一个重要阶段。在这一阶段中，得到充分发展的数据结构和算法丰富了计算机科学，为数据管理技术的进一步发展打下了基础，现在仍是计算机软件科学的重要基础。文件系统中的文件是为某一特定应用服务的，文件的逻辑结构对该应用程序来说是优化的，因此要想对现有的数据再增加一些新的应用会很困难，系统不容易扩充。一旦数据的逻辑结构改变，必须修改应用程序，修改文件结构的定义。应用程序的改变，例如程序改变不同的高级语言等，也将引起文件的数据结构的改变。因此数据与程序之间仍缺乏独立性。可见，文件系统仍然是一个不具备弹性的无结构数据集合，即文件之间是孤立的，不能反映现实世界事物之间的内在联系。在文件系统阶段，程序与数据之间的关系如图 1.2 所示。

3. 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，同时为了让用户能更方便地使用这些数据资源，在计算机科学中，逐渐形成了数据管理这一独立分支。计算机中的数据及数据的管理统一由数据库系统来完成。为此，数据管理技术进入数据库系统阶段。

数据库系统克服了文件系统的缺陷，提供了对数据更高级、更有效的管理。这个阶段的程序和数据的联系通过数据库管理系统（DBMS）来实现，如图 1.3 所示。

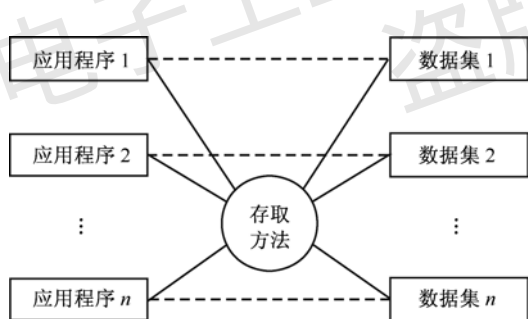


图 1.2 文件系统阶段应用程序与数据之间的对应关系

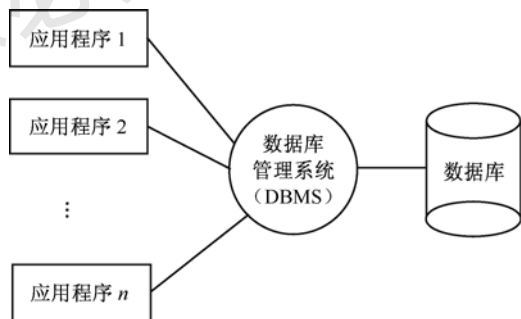


图 1.3 数据库系统阶段应用程序与数据之间的对应关系

概括起来，数据库系统阶段的数据管理具有以下特点：

(1) 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实现。通过所有存取路径表示自然的数据联系是数据库与传统文件的根本区别。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。

(2) 有较高的数据独立性

数据的逻辑结构与物理结构之间的差别可以很大。用户以简单的逻辑结构操作数据而无须考虑数据的物理结构。数据库的结构分成用户的局部逻辑结构、数据库的整体逻辑结构和物理结构三级。用户的数据和外存中的数据之间转换由数据库管理系统实现。

(3) 数据库系统为用户提供了方便的用户接口

用户可以使用查询语言或终端命令操作数据库, 也可以用程序方式操作数据库。

(4) 数据库系统提供了数据控制功能

数据库系统提供了数据控制功能例如:

- 数据库的并发控制, 对程序的并发操作加以控制, 防止数据库被破坏, 杜绝提供给用户不正确的数据。
- 数据库的恢复, 在数据库被破坏或数据不可靠时, 系统有能力把数据库恢复到最近某个正确状态。
- 数据完整性, 保证数据库中数据始终是正确的。
- 数据安全性, 保证数据的安全, 防止数据的丢失、破坏。

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(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, 而导致程序重写和数据结构重新变动。

综上所述, 这三个阶段的特点及其比较如表 1.5 所示。

表 1.5 数据管理的发展阶段

		人工管理阶段	文件系统阶段	数据库系统阶段
背景	应用背景	科学计算	科学计算、管理	大规模管理
	硬件背景	无直接存取存储设备	磁盘、磁鼓	大容量磁盘
	软件背景	没操作系统	有文件系统	有数据库管理系统
	处理方式	批处理	联机实时处理、批处理	联机实时处理、分布处理、批处理
特点	数据的管理者	用户	文件系统	数据库管理系统
	数据面向的对象	某一应用程序	某一应用	现实世界
	数据的共享程度	无共享, 冗余度极大	共享性差, 冗余度大	共享性高, 冗余度小
	数据的独立性	不独立, 完全依赖于程序	独立性差	具有高度的物理独立性和一定的逻辑独立性
	数据的结构化	无结构	记录内有结构、整体无结构	整体结构化, 用数据模型描述
	数据控制能力	应用程序自己控制	应用程序自己控制	由数据库管理系统提供数据安全性、完整性、并发控制和恢复能力

1.2.3 数据库系统的特点

与人工管理和文件系统相比, 数据库系统具有如下特点:

(1) 数据结构不是面向单一的应用, 而是面向全组织

以医院管理为例, 要想避免数据冗余和数据程序之间的依赖性, 就要将医生和病人两类不同的数据之间彼此建立关系。当需要增加新的应用, 比如药品管理, 则只要再增加新的联系。这种思想只是数据库方法的雏形, 从文件内部的记录的结构化, 扩大到不同的文件记录之间建立一种联系。但是它还有局限性, 因为它还是从应用的角度去看待数据, 还应进一步从整个组织的数据结构考虑。假设所考虑的这个组织——医院, 就还应该有医院科室信息、诊断信息、收费信息等。不同应用考虑的是整个数据集合的某个有用的子集, 整个组织的数据是结构化的。这样描述数据时不仅描述数据本身, 还要描述数据之间的联系。数据的结构化是数据库主要特征之一, 这是数据库与文件系统的根本区别。至于这种结构化是如何实现的, 则与数据库系统采用的数据模型有关。

(2) 数据冗余小, 易扩充

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。

(3) 数据独立于程序

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。举例来讲, 在进行病人诊断信息管理时, 医生等数据来自于数据的医生部分, 病人信息来自于数据的病人部分, 经过映像组成局部的诊断信息, 由数据库维持这种映像。当总体的逻辑结构改变时, 比如医生和病人数据的结构发生了变化, 数据库为这种改变建立一种新的映像, 可以保证局部数据——诊断信息的逻辑结构不变, 程序是面向这个局部数据的, 所以程序就无须改变。

(4) 统一的数据管理功能

统一的数据管理功能, 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。

① 数据库的安全性是指防止非法用户的非法使用数据库而提供的保护。比如, 不是医院的成员不允许使用医院管理系统, 病人允许读取诊断信息但不允许修改诊断信息等。

② 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。比如在医院管理系统中, 数据库管理系统必须保证输入的收费值大于 0, 否则, 系统发出警告。

③ 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外存的数据库中, 实现物理的改变。

由于数据库的这些特点, 它的出现使信息系统的研制从围绕加工数据的程序为中心转变到围绕共享的数据库来进行。便于数据的集中管理, 也提高了程序设计和维护的效率。提高了数据的利用率和可靠性。当今的大型信息管理系统均是以数据库为核心的。数据库系统是计算机应用中的一个重要阵地。总之, 数据库技术正是研究如何科学地组织和储存数据, 如何高效地获取和处理数据。数据库技术是到目前为止发展成熟的数据管理的最新技术。



1.3 数据模型

对于模型, 人们并不陌生。一张地图, 一组建筑设计沙盘, 一架精致的航模飞机都是具体的模型, 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, 它是对现实世界数据特征的抽象, 即是用来描述数据、组织数据和对数据进行操作的。由于计算机不可能直接处理现实世界中的具体事物, 所以人们必须先要把具体事物转换成计算机能够处理的数据。

现有的数据库系统均是基于某种数据模型的。数据模型是数据库系统的核心和基础。

根据模型应用的不同目的, 可以将模型划分概念模型、数据模型和物理模型。其中, 物理模型主要是描述数据在计算机中存储的具体细节, 表示能由 DBMS 理解的概念模式。物理模型提供的概念

通常对计算机专家很有意义,但对于一般最终用户则没什么用处,因此在这里我们就不加阐述了。下面简要介绍概念模型和数据模型。

1.3.1 概念模型

概念模型也称信息模型,是按用户的观点对数据和信息建模,是现实世界到机器世界的一个中间层次。

概念模型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,是数据库设计人员进行数据库设计的有力工具,也是数据库设计人员与用户之间进行交流的语言。因此,概念模型一方面要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,另一方面应该简单、清晰、易于理解。

概念模型主要涉及一些基本的概念(如实体、属性、联系等)和其相应的表示方法,即实体-联系图。实体-联系图中:

- 实体:用矩形表示,矩形框内写明实体名。
- 属性:用椭圆形表示,并用无向边将其与相应的实体连接起来。
- 联系:用菱形表示,菱形框内写明联系名,并用无向边分别与有关实体连接起来,同时在无向边旁标上联系的类型(1:1, 1:n 或 m:n)。

1.3.2 数据模型

数据模型即逻辑数据模型,它是按照计算机系统的观点对数据建模,表示独立于 DBMS 的逻辑视图,隐藏了数据存储的一些细节,主要用于 DBMS 的实现。它是数据库系统的核心和基础。各种机器上实现的 DBMS 软件都是基于某种数据模型或者说是支持某种数据模型的。

一般地讲,数据模型是严格定义的一组概念的集合,它描述了系统的静态特征、动态特征和完整性约束条件。因此,数据模型通常由数据结构、数据操作和完整性约束三部分组成。

目前,数据库领域中常用的数据模型主要包括层次模型、网状模型、关系模型和面向对象模型等。其中,层次模型和网状模型统称为格式化模型。

格式化模型的数据库系统在 20 世纪 70 年代至 80 年代初非常流行,在数据系统产品中占据了主导地位,现在已逐渐被关系模型的数据库系统取代。20 世纪 80 年代以来,面向对象的方法和技术在计算机各个领域,包括程序设计语言、软件工程、信息系统设计、计算机硬件设计等各方面都产生了深远的影响,也促进了数据库面向对象数据模型的研究和发展。许多关系数据库厂商为了支持面向对象模型,对关系模型做了扩展,从而产生了对象关系模型。

1. 层次模型

层次模型是数据库系统中最早出现的数据模型。用树型(层次)结构表示实体类型及实体间联系的数据模型称为层次模型(Hierarchical Model)。

层次模型是指用树型结构表示实体及其之间的联系,树中每一个结点代表一个记录类型,树状结构表示实体型之间的联系。在一个层次模型中的限制条件是:树的根结点有且仅有一个结点,无父结点;其他结点有且仅有一个父结点。

2. 网状模型

用有向图结构表示实体类型及实体间联系的数据结构模型称为网状模型(Network Model)。

在数据库中,把满足以下两个条件的基本层次联系集合称为网状模型:

- (1) 允许一个以上的结点无双亲;
- (2) 一个结点可以有多个的双亲。

网状模型是一种比层次模型更具普遍性的结构，它去掉了层次模型的两个限制，允许多个结点没有双亲结点，允许结点有多个双亲结点，此外它还允许两个结点之间有多种联系（称之为复合联系）。因此网状模型可以更直接地去描述现实世界。而层次模型实际上是网状模型的一个特例。

3. 关系模型

关系模型是目前最重要的一种数据模型，它是用二维表的形式表示实体和实体间联系的数据模型。

关系数据模型是以集合论中的关系概念为基础发展起来的。关系模型中无论是实体还是实体间的联系均由单一的结构类型——关系来表示。在实际的关系数据库中的关系也称表。一个关系数据库就是由若干个表组成。

关系模型由关系数据结构、关系操作集合和关系的完整性约束组成。

4. 面向对象模型

面向对象模型是捕获在面向对象程序设计中支持的对象语义的逻辑数据模型，它是持久的和共享的对象集合，具有模拟整个解决问题的能力。面向对象数据模型把实体表示为类，一个类描述了对象属性和实体行为。但和实体不同的是，一个对象所包含的信息不只是对象内部事实之间的联系，也包括它和其他对象的联系。因此，对象里的事实被赋予了更多的语义。面向对象数据库通过逻辑包含来维护联系，它是面向对象概念与数据库技术相结合的产物。

第 2 章将对各种模型进行详细介绍。



1.4 数据库系统的结构与组成

数据库系统这个术语指的是在一个数据库环境下，用来定义和控制数据的收集、存储、管理和使用的各个成分的总和。

考察数据库系统的结构可以有多种不同的角度：从数据库最终用户的角度看，数据库系统的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/数据库服务器多层结构等，这是数据系统外部的体系结构；从数据库管理系统的角度看，数据库系统通常采用三级模型结构，这是数据库管理系统内部的系统结构。



1.4.1 数据库系统的体系结构

目前，常用的数据库系统的体系结构主要有客户-服务器（C/S）结构和浏览器/服务器（B/S）结构两种。

（1）客户-服务器结构

客户-服务器（C/S, Client/Server）结构是目前流行的数据库系统结构。在这种结构中，客户机提出请求，服务器对客户机的请求做出回应。C/S 结构的本质在于通过对服务功能的分布，实现分工服务。每一个服务器都为整个局域网系统提供共享服务，供所有客户机共享；客户机上的应用程序借助于服务器的服务功能以实现复杂的应用功能。在 C/S 结构中，数据存储层处于服务器上，应用层和用户界面层处于客户机上。客户机负责管理用户界面，接收用户数据，处理应用逻辑，生成数据库服务请求，将该请求发送给服务器，同时接收服务器返回的结果，并将结果按一定格式显示给用户。

（2）浏览器-服务器结构

浏览器-服务器（Browser/Server, B/S）结构是针对 C/S 的不足而提出的。在 B/S 结构中，客户机端仅安装通用的浏览器软件，实现用户的输入/输出，而应用程序不再是安装在客户机端，而是在服务器端安装与运行。在服务器端，除了要有数据库服务器保存数据并执行基本的数据库操作外，还要

有另外的称为应用服务器的服务器处理客户端提交的处理要求。也就是说，C/S 结构中客户端运行的程序已转移到应用服务器中，此时的客户机可称为“瘦客户”。应用服务器充当了客户机与数据库服务器的中介，架起了用户界面与数据库之间的桥梁，所以也称为三层结构。

1.4.2 数据库系统的模式结构

早在 1971 年，由数据系统语言会议（CODASYL，1971）任命的 DBTG（数据库任务组），提出了关于数据库系统的标准术语和一般体系结构的规范。DBTG 认为需要一个两层的方法，即从系统角度看的模式（schema）和从用户角度看的子模式（subschema）。美国国家标准化协会（ANSI）标准规划和需求委员会（SPARC）在 1975 年提出了一个类似的术语和体系结构，ANSI/X3/SPARC。ANSI-SPARC 提出了带系统目录的三层方法。

为此，在以后的报告中首先标识三层抽象，即描述数据的三个不同层次。这些层次形成了一个三层体系结构，包括外部层、概念层和内部层。这些分层方法能够实现以下 3 个特点：

- ① 程序与数据分离（程序与数据独立性和程序与操作独立性）。
- ② 支持多用户视图。
- ③ 使用编目存储数据库描述（模式）。三层体系结构实现用户的数据库视图与数据库的物理描述分离。

分层的原因是：

- ① 每个用户应该能够访问相同的数据，但用各自自定义的数据视图。每一个用户都应该能够改变数据视图，但这些改变不应该影响其他的用户。
- ② 不应该要求用户直接处理数据库物理存储的细节，例如索引。
- ③ 数据库管理员（DBA）应该能够在不影响用户视图的情况下，修改数据库存储结构。

下面讨论数据库系统三层模式体系结构（three-schema architecture），三层体系结构对实现上述特点非常重要。

1. 模式的概念

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说明，值是型的一个具体赋值。例如，医生记录定义为（医生编号，姓名，性别，年龄，技术职称）这样的记录型，而（161，刘景，男，66，主任医师）则是该记录型的一个记录值。

所谓模式，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。

模式是相对稳定的，而实例是相对变动的，因为数据库中的数据是在不断更新。模式反映的是数据的结构及联系，而实例反映的是数据库某一时刻的状态。

2. 三层模式结构

如图 1.4 所示，三层模式体系结构的目的是将用户与物理数据库分离开来。在这个体系结构中，可以将模式分为如下 3 个层次。

（1）内部层有一个内模式（Internal Schema），它描述的是数据库的物理存储结构。

内模式也称存储模式，一个数据库只有一个内模式。它使用一个物理数据模型，全面描述数据库中数

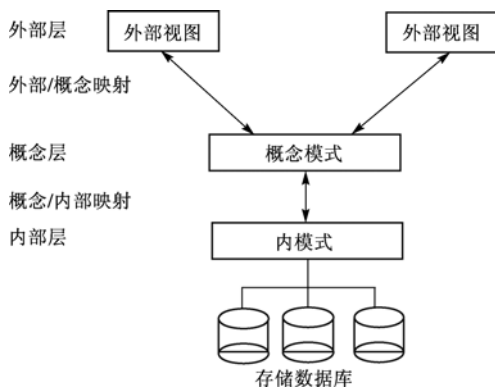


图 1.4 三层模式体系结构

据存储的全部细节和存取路径。它与操作系统接口，以便将数据存放到存储设备上，建立索引、检索数据等。内部层与以下工作相关：

- 数据和索引的存储空间分配；
- 用于数据存储的记录描述；
- 记录放置；
- 数据压缩和数据加密技术。

内部层之下是物理层，物理层可能在 DBMS 的指导下受操作系统的控制。然而，DBMS 和操作系统在物理层上的功能分割并不是十分清晰，并且因系统而异。一些 DBMS 充分利用操作系统的访问方法，而另外一些只使用一些最基本的功能，然后创建他们自己的文件组织机制。处于 DBMS 之下的物理层所包含的内容只有操作系统掌握。

(2) 概念层有一个概念模式 (Conceptual Schema)，它为用户群体描述了整个数据库的结构。

概念模式也简称模式或逻辑模式，它隐藏了物理存储结构的细节，并重点描述实体、数据类型、关系、用户操作和约束。通常，实现了一个数据库系统之后，会使用一个表示数据模型来描述概念模式。这种概念模式通常以高级数据模型中的一个概念模式设计为基础。概念层描述如下：

- 所有的实体、实体的属性和实体间的联系；
- 数据的约束；
- 数据的语义信息；
- 安全性和完整性信息。

概念层支持所有外部视图，只要对用户可访问的数据包含在概念层或者由概念层可导出。然而，这层一定不能包括任何依赖于存储的详细信息。例如，对于实体的描述只能包括属性的数据类型（如整型、实型、字符型等）及它们的长度（如字符或者数字的最大位数），但是不能包括任何与存储相关的信息。

(3) 外部层包括许多外模式 (External Schema)。

外模式也称作子模式。每个外模式描述的是一个特定用户组所感兴趣的那部分数据库，而对该用户组隐藏了数据库的其他部分。各个外模式通常都用一个表示数据模型实现，并且可能以高级数据模型的一个外模式设计为基础。

对于同一个数据，不同的视图可能会有不同的表达方式。例如，一些用户可能是以日、月、年的形式查看日期，而另外一些用户可能是以年、月、日的形式查看日期。一些视图可能包含导出的或者通过计算得到的数据，数据实际上没有存储到数据库中，而是当需要时才产生的。

三层模式体系结构是一个非常方便的工具，通过这个结构，用户就可以直观地了解数据库系统中的模式层次。大多数 DBMS 并不是将这三层模式完全分离开来，而只是在一定程度上支持三层模式体系结构。有些 DBMS 可能在概念模式中还包括一些物理层的细节。在大多数支持用户视图的 DBMS 中，外模式也是用描述概念层信息的同一数据模型来指定的。

3. 模式间的映射与数据独立性

数据库管理系统负责三类模式之间的映射，同时检查模式以确保一致性。DBMS 必须检查每个外模式都由概念模式导出，并且它必须使用概念模式中的信息，以完成内、外模式的映射。概念模式通过概念层到内部层的映射与内部模式相联系。这样，DBMS 就能在物理存储中找出构成概念模式中逻辑记录的实际记录或者记录的组合，以及对逻辑记录进行操作过程中应执行的约束。它还允许两类模式在实体名称、属性名称、属性顺序、数据类型等方面存在不同。最后，每一个外部模式通过外部层到概念层的映射与概念模式相联系。这就允许 DBMS 将用户视图中的名称映射到概念模式中相应的部分。

图 1.5 给出了一个不同层的例子。存储两个不同的关于医生详细信息的外部视图：一个包含医生编号 (Sno)、名字 (Sname)、性别 (Dsex) 和年龄 (Dage)，另一个包含医生编号 (Dno)、所在部门编号 (Ddeptno)、技术级别 (Dlevel) 和工资 (Dsalary)。这些外部视图归并为一个概念视图。此归并的主要不同之处是将 Sno 和 Sname 映射为 Dno 和 Dname。DBMS 维护外部视图到概念层的映射。概念层则被映射到内部层，在内部层中包含对于概念记录结构的一个物理描述。在这一层，可以看到一个用高级语言定义的结构。这个结构包含一个指针 next，靠这个指针医生记录在物理上连接成一条链。DBMS 同样要维护概念层到内部层的映射。

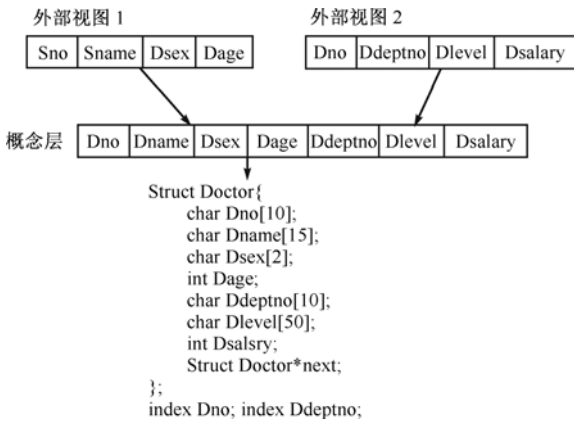


图 1.5 三个层次的不同之处

区分数据库的描述和数据库本身是很重要的。数据库的描述是数据库模式。这个模式是在数据库设计过程中指定的，并且不会经常改变。然而，数据库中的实际数据可以经常改变。例如，每次插入一个医生或病人信息时，它都会发生改变。

三层体系结构的一个主要目的是保证数据的独立性，这意味着对较低层的修改不会对较高层造成影响。数据的独立性分为逻辑独立性和物理独立性两类。

① 逻辑数据独立性是指外部模式不受概念模式变化影响。对概念模式的修改，例如新实体、属性或者联系的添加或删除，应该不影响已存在的外部模式，也不需要重新编写应用程序。显然，重要的修改只应由相关的用户知道，其他的用户不必知道。

② 物理数据独立性是指概念模式不受内部模式变化的影响。对内部模式的修改，例如使用不同的文件组织方式或者存储结构，使用不同的存储设备，修改索引或散列算法，应该不影响概念模式和外部模式。对用户来讲，唯一要注意的是对性能的影响。实际上，性能变坏是改变内部模式最常见的原因。

1.4.3 数据库系统的一般组成

如图 1.6 所示，数据库系统由五部分组成：硬件系统、数据库集合、数据库管理系统及相关软件、数据库管理员和用户。

1. 硬件系统

运行数据库系统的计算机需要有足够大的内存、足够容量的磁盘等联机直接存取设备和较高的通道能力，以及支持对外存的频繁访问，还需要足够数量的脱机存储介质，如软盘、光盘、磁带等存放数据库的备份。

2. 数据库集合

系统包括若干个设计合理、满足应用需要的数据库。

3. 系统软件

软件指的是数据库系统中被计算机使用的程序的集合。需要三种类型的软件来实现数据库系统的全部功能：操作系统、DBMS 软件和应用程序。

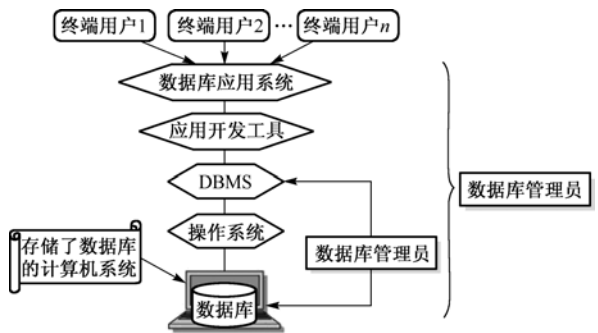


图 1.6 数据库系统的组成

① 操作系统软件管理所有的硬件并确保所有的其他软件能够在计算机上运行。操作系统软件的实例包括微机上运行的磁盘操作系统 (DOS)、OS/2 和 WINDOWS, 小型机上运行的 UNIX 和 VMS, IBM 大型机上运行的 Z/OS。

② DBMS 软件管理在数据库系统范围内的数据库。一些 DBMS 软件包括微软公司的 Access 和 SQL Server、Oracle 公司的 Oracle、IBM 公司的 DB2 等。

③ 应用程序用来访问和操作 DBMS 里的数据, 并且管理对数据访问和操作的计算机环境。应用程序大多数用来访问从数据库中获取的数据, 生成报告、表格和其他辅助决策的信息。

4. 数据库管理员 (DBA, Database Administrator)

数据库管理系统一般需要专人来对数据库进行管理, 这个人称为数据库管理员 (DBA)。数据库管理员负责数据库系统建立、维护和管理。数据库管理员的职责包括: 定义并存储数据库的内容、监督和控制数据库的使用、负责数据库的日常维护, 必要时重组或改进数据库。

5. 用户

数据库系统必然涉及不同的用户。数据库的用户分为两类: 一类是最终用户, 主要对数据库进行联机查询或者通过数据库应用系统提供的界面来使用数据库。这些界面包括菜单、表格、图形和报表。另一类是专业用户, 即应用程序员, 他们负责设计应用系统的程序模块, 对数据库进行操作。



1.5 数据库语言

数据库系统提供了 4 种不同类型的语言: 数据查询语言 (Data Query Language)、数据定义语言 (Data Definition Language)、数据操纵语言 (Data Manipulation Language) 和数据控制语言 (Data Control Language)。其中, 数据查询语言主要用来查询数据; 数据定义语言用于定义、撤销和修改数据模式, 如表、视图、索引; 数据操纵语言用于增、删、改数据; 数据控制语言用于数据访问权限的控制。

1.5.1 四大数据库语言

1. 数据查询语言 (DQL)

数据库查询语言是用户用于从数据库中请求获取信息的语言。它可以分为过程化语言和非过程化语言。在过程化语言中, 用户指导系统对数据库执行一系列操作以计算所需结果。在非过程化语言中, 用户只需描述所需信息, 而不用给出获取该信息的具体过程。

目前, 有很多商业性的或者实验性的数据查询语言, 如现在使用最广泛的结构化查询语言。

2. 数据定义语言 (DDL)

数据定义语言用来定义数据的结构, 如创建、修改或者删除数据库对象。

3. 数据操纵语言 (DML)

用户通过数据操纵语言可以实现对数据库中的数据进行追加、插入、修改、删除、检索等操作。主要包括: 向数据库中插入新的信息、从数据库中删除信息和修改数据库中存储的信息。

不同的 DBMS 语言的语法格式也不相同, 以其实现方式可分为两个类型: 一类是 DML 可以独立交互使用, 不依赖于任何程序设计语言, 称为自主型语言; 另一类是宿主语言 DML, 嵌入到宿主语言中使用, 如嵌入 C、JAVA 等程序设计语言中。在使用高级语言编写应用程序时, 当需要调用数据库中的数据时, 则要用 DML 语句来操纵数据。

4. 数据控制语言(DCL)

数据控制语言用来授予或回收访问数据库的某种权限,并控制数据库操纵事务发生的时间及效果,对数据库实行监视等。

1.5.2 结构化查询语言 SQL

结构化查询语言 SQL (Structured Query Language) 是关系数据库的标准语言。它是一种高级的非过程化编程语言,允许用户在高层数据结构上工作。它不要求用户指定对数据的存放方法,也不需要用户了解具体的数据存放方式,所以具有完全不同的底层结构。不同数据库系统可以使用相同的 SQL 语言作为数据输入与管理的接口。它以记录集合作为操作对象,所有 SQL 语句接受集合作为输入,返回集合作为输出。这种集合特性允许一条 SQL 语句的输出作为另一条 SQL 语句的输入,所以 SQL 语句可以嵌套,这使它具有极大的灵活性和强大的功能。在多数情况下,在其他语言中需要一大段程序实现的功能,而在 SQL 语言中只需要一条 SQL 语句就可以达到目的,这也意味着用 SQL 语言可以写出非常复杂功能的语句。

SQL 集数据查询、数据定义、数据操纵和数据控制功能于一体,是一个通用的、功能极强的关系数据库语言。它包含 4 个部分:

(1) 数据查询语言(DQL)

例如: SELECT 语句。

(2) 数据定义语言(DDL)

例如: CREATE、DROP、ALTER 等语句。

(3) 数据操作语言(DML)

例如: INSERT (插入)、UPDATE (修改)、DELETE (删除) 语句。

(4) 数据控制语言(DCL)

例如: GRANT、REVOKE、COMMIT、ROLLBACK 等语句。

本书第 3 章将对 SQL 进行更为详尽的介绍。



1.6 数据库系统的发展趋势及前沿技术

随着数据库技术的迅速发展以及数据库管理系统的广泛应用,原有的数据库系统很难适应新的应用领域中的复杂对象和这些对象的复杂行为的需求,如传统数据模型不支持用户自定义数据类型、不支持异构数据等。新的应用需求推动了数据库技术的发展,主要有以下 3 个发展趋势。

(1) 趋势一: 对于异构数据的支持

在以前的数据库中,一直强调数据的标准化。也就是说,在以前,数据库主要管理的是结构化的数据,数据是以行与列的二维形式进行存储。若要实现三维数据,如数据仓库等,都要根据一定的规则对数据库进行叠加才能够完成。但是,这个工作量会很大,而且维护起来也会很吃力。

随着企业系统集成的需要,企业想把电子邮件、多媒体文件、附件等都包含在一个系统中,以减少系统的重复投资。在这种趋势下,支持异构数据的数据库技术也纷纷出现,主要是为了让异构数据能够像结构化数据那样进行管理和查询等。

因此,从专业的角度讲,在目前数据库技术的发展中,最突出的变化就是从二维表存储方式到多维数组的存储方式的转变。通过多维数组技术,不但可以把语音文件等异构数据存入到数据库中,便于统一管理,而且还可以直接实现数据仓库。

(2) 趋势二：对于网络计算的支持

网络运算技术可以让用户更好的在网络环境中分享存储资源，并且同时可以保障数据在安全方面的需求。网络数据库的最大的优势在于，数据库可以利用这个技术，将一个数据库应用部署在多台独立的服务器中，实现一个高容错的运算平台，以提高数据库应用的稳定性，减少数据库宕机的时间。这对于一些大型的数据库应用，如银行的数据库系统，具有非常现实的意义。

(3) 趋势三：管理的智能化

如何让数据库系统自动优化资源用量。数据库的性能一直是数据库管理员所关心的问题。在以前的数据库系统中，数据库管理员最重要的任务就是监控数据库服务器的性能，并对其进行及时的调整，让数据库服务器达到最好的运行状态。这项工作往往需要耗费数据库管理员大量的时间与精力。不过这种状况将会在不久的将来得到彻底的改变。SQL Server 数据库将会在新版本的数据库中加入很多的自我调整功能。例如，自我调整内存分配，让数据库负责优化内存分派，以提高数据库性能。因为我们都知，数据库系统在内存中读取数据要比在硬盘中读取数据快成百上千倍。让数据库根据内存的使用情况，合理分配内存，可以提高服务器内存的使用率。从而达到提高数据库性能的目的。

在将来，数据库厂商为了获得市场，在关注业务需求的同时，会更多地关注如何降低管理成本，如何实现管理的智能化，从而得到数据库工程师的认同。

针对上面的发展趋势，分别产生了下列新的数据库技术。

(1) XML 数据库

从近几年看，各大数据库厂商几乎无一例外地在数据库内支持 XML (Extensible Markup Language, 可扩展的置标语言)。所谓 XML 是定义文档结构的机制，XML 规范定义了一个对文档进行标记的标准。目前，XML 已是各种数据特别是文档的首选格式，国际主流的数据库厂商们自然也随行就市，全都推出了兼容传统关系型数据与层次型数据 (XML 数据) 混合应用的新一代数据库产品。

甲骨文早在 Oracle8i 中就支持 XML，在 Oracle9i 提供了 XmlDB 功能，支持 XMLType 数据类型，并在 Oracle10g 进一步加强了功能和提高了性能，如 XQuery 的性能。

IBM DB2 9 中新 XML 支持包括纯 XML 存储、XML 索引、XQuery 和高级的 XML 模式处理。这与商业关系数据库中以前的任何技术都不同。

微软 SQL Server 2005 为 XML 数据处理提供了广泛的支持。XML 值可以自然地存储在 XML 数据类型列中，并可以根据 XML 架构集合进行类型化，或者保持非类型化。

Sybase 公司的 ASE 数据库与其嵌入式数据库都已经支持 XML，可以从关系查询中创建 XML 格式的结果集，并且各具特色。

(2) 网络计算

商业计算的需求使用户需要高性能的计算方式，而超级计算机的价格却阻挡了高性能计算的普遍能力。于是造价低廉而数据处理能力超强的计算模式——网络计算应运而生。

网络计算的主要优势在于：

- ① 共享资源，将可用资源汇集起来形成共享池；
- ② 虚拟化堆栈的每一层，可以如同管理一台计算机一样管理资源；
- ③ 基于策略实现自动化负载均衡。

从 IT 行业的趋势来看，企业也正在向网络计算转移，这在很大程度上是低成本的单片服务器驱动的。同时，共享存储技术也简化了硬件的虚拟化和供应，硬件供应商也开始提供实现硬件虚拟化和供应的管理软件。



(3) BI 商业智能数据库

为应对日益加剧的商业竞争,企业不断增加内部 IT 及信息系统,使企业的商业数据成几何数量级不断递增。如何能够从这些海量数据中获取更多的信息,以便分析决策将数据转化为商业价值,就成为目前数据库厂商关注的焦点,这其中离不开商业智能(BI, Business Intelligence)。

而从用户对数据管理需求的角度看,可分为两类:联机事务处理(OLTP, Online Transaction Processing)应用、联机分析处理(OLAP, Online Analysis Processing)与辅助决策支持(DSS, Decision Support System)两大类,也就是说,数据库不仅要支持 OLTP,还应该为业务决策、分析提供支持。目前,主流的数据库厂商都已经把支持 OLAP 和商业智能作为关系数据库发展的一大趋势。

从国内市场需求来看,企业信息化发展正处于一个系统集成阶段,因此对高端智能决策系统的需求增强。目前,金融、保险领域的国有企业,大部分已经建立了部门级的数据集市。今后,这些企业中企业级的 BI 系统将会逐步建立。有些发达省份的企业,已经初步建立 BI 系统,或者正在考虑建设 BI 系统。而对国外企业而言,这一趋势就更加明显了。因此, BI 将会是未来发展的重中之重。



1.7 习题

1. 什么是数据库?有什么特点?
2. 什么是数据库系统?主要由几个部分组成?简要解释各组成部分的含义。
3. 数据库管理系统的功能有哪些?
4. 数据管理技术的三个发展阶段是什么?
5. 试述数据库系统的优点。
6. 试述数据库系统的三级模式结构。
7. 解释数据与程序的物理独立性和逻辑独立性。为什么数据库系统具有数据与程序的独立性?
8. DBA 的职责是什么?
9. 自行虚拟设计一个关系数据库系统的实例,说明其可能的应用环境、希望实现的功能。