

# 数据库原理及应用

## — *SQL Server 2000*

夏冰冰 主编



国防工业出版社

National Defense Industry Press

000755096

# 数据库原理及应用

## ——*SQL Server 2000*

夏冰冰 主编

国防工业出版社

·北京·

## 内 容 简 介

本书主要介绍关系数据库的基本理论及关系数据库的理论基础——关系代数的知识;通过 SQL Server 2000 介绍一个具体的关系数据库产品的各种操作,并通过图例进行说明。Transact - SQL 语言是本书的重点,主要包括数据定义、数据查询和数据更新三大部分。在书中还将介绍范式及关系数据库的规范化,这是数据库设计的理论基础,数据库规范化程度以第一范式、第二范式、第三范式和 BC 范式为主。数据库设计这一章以具体实例介绍数据库设计的各个步骤,本书的最后还将以具体的数据库应用程序为例介绍数据库应用程序的开发过程。书中使用的例子是学生课程数据库。

本书可作为计算机专业或非计算机专业本科生的教材,也可作为从事计算机应用开发的各类人员的参考书。

### 图书在版编目(CIP)数据

数据库原理及应用:SQL Server 2000 / 夏冰冰主编. —北京:  
国防工业出版社,2009. 12  
ISBN 978 - 7 - 118 - 06605 - 0

I. ①数... II. ①夏... III. ①关系数据库 - 数据库管  
理系统,SQL Server 2000—教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2009)第 240879 号

※

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号 邮政编码 100048)

天利华印刷装订有限公司印刷

新华书店经售

\*

开本 787 × 1092 1/16 印张 13½ 字数 310 千字  
2009 年 12 月第 1 版第 1 次印刷 印数 1—4000 册 定价 26.00 元

(本书如有印装错误,我社负责调换)

国防书店:(010)68428422  
发行传真:(010)68411535

发行邮购:(010)68414474  
发行业务:(010)68472764

# 前 言

自 20 世纪 60 年代中期以来,数据库技术得到了飞速的发展,目前在各个应用领域中得到了广泛的使用,对高等院校来说,数据库原理与应用也成为一门重要的主干课程。本书根据作者多年的讲授经验,结合高等院校教学实际,将数据库基本理论与 SQL Server 的应用结合起来,在书中进行了以下几点改进:

(1) 由于层次数据库和网状数据库已经很少使用,因此在本书中不再涉及这方面的内容。

(2) 对关系数据库的运算只介绍关系代数,不再介绍关系演算。

(3) 本书的应用部分结合 SQL Server 来介绍,各个章节根据教学实际来组织,按照由浅入深、层层深入的原则,结合丰富的实例及介绍,形式多样,通俗易懂。

(4) 将数据的完整性的理论及 SQL Server 对完整性的支持在第 5 章中介绍,理论结合实际,体系完整,编排合理。

(5) 在第 7 章中只介绍函数依赖,对多值依赖及第四范式、第五范式不再介绍。

全书内容共 10 章,其中第 1 章为数据库系统概述,第 2 章为关系数据库的基本知识,第 3 章为 SQL Server 2000 简介和基本操作,第 4 章为 Transact - SQL 语言,第 5 章为数据完整性,第 6 章为 SQL 编程和存储过程,第 7 章为关系数据库设计规范化,第 8 章为数据库设计,第 9 章为数据库安全性,第 10 章为数据库应用。

本书由夏冰冰主编。参与本书编写的有:夏冰冰(第 2 章、第 4 ~ 5 章、第 7 ~ 8 章),张岳(第 3 章、第 6 章、第 9 章),田睿(第 1 章和第 10 章),夏旻和夏冰冰负责全书组织和统稿。

本书可作为计算机专业或非计算机专业本科生教材,也可作为计算机应用开发的人员使用。

作者可提供本书的电子课件,如需要,可与作者([jennifer\\_xiababy@yahoo.com.cn](mailto:jennifer_xiababy@yahoo.com.cn))或出版社联系。

由于作者水平有限,书中难免有不当之处,还望读者指正。

编 者  
2009 年 9 月

# 目 录

<b>第 1 章 数据库概述</b> .....	1
1.1 数据库系统概述 .....	1
1.1.1 数据库基本概念 .....	1
1.1.2 数据库系统的特点 .....	2
1.2 数据模型 .....	4
1.2.1 概念数据模型 .....	4
1.2.2 逻辑数据模型 .....	5
1.3 数据库系统结构 .....	6
1.3.1 数据库外部的系统结构 .....	7
1.3.2 数据库内部的系统结构 .....	9
习题 .....	10
<b>第 2 章 关系数据库的基本知识</b> .....	11
2.1 关系数据结构及形式化定义 .....	11
2.1.1 关系 .....	11
2.1.2 关系模式 .....	13
2.1.3 关系数据库 .....	14
2.2 关系代数 .....	14
2.2.1 传统的集合运算 .....	15
2.2.2 专门的关系运算 .....	17
2.3 关系完整性 .....	21
2.3.1 几个概念 .....	21
2.3.2 关系的完整性 .....	23
习题 .....	24
<b>第 3 章 SQL Server 2000 简介和基本操作</b> .....	25
3.1 SQL Server 2000 概述 .....	25
3.2 SQL Server 2000 的安装 .....	26
3.3 SQL Server 2000 的系统组成 .....	31
3.3.1 SQL Server 2000 服务器组件 .....	32
3.3.2 服务器启动、暂停和停止 .....	33
3.3.3 SQL Server 2000 主要的管理工具 .....	33
3.4 创建数据库和表 .....	36
3.5 数据查询和创建视图 .....	38

3.6	数据更新	43
3.7	数据导入/导出	45
	习题	48
<b>第4章</b>	<b>Transact - SQL 语言</b>	<b>49</b>
4.1	Transact - SQL 语言概述	49
4.2	数据定义	50
4.2.1	数据库的定义	50
4.2.2	基本表的定义	56
4.2.3	索引的定义	59
4.2.4	视图的定义	62
4.3	数据查询	66
4.3.1	单表查询	67
4.3.2	连接查询	80
4.3.3	嵌套查询	84
4.3.4	集合查询	92
4.4	数据更新	92
4.4.1	插入数据	93
4.4.2	修改数据	94
4.4.3	删除数据	95
	习题	96
<b>第5章</b>	<b>数据完整性</b>	<b>98</b>
5.1	数据完整性概述	98
5.2	企业管理器中设计数据完整性	99
5.2.1	使用约束	99
5.2.2	规则	109
5.2.3	默认值	113
5.2.4	标识列	116
5.3	使用 Transact - SQL 设计数据完整性	117
5.3.1	使用约束	117
5.3.2	使用规则	122
5.3.3	使用默认值	122
5.3.4	使用标识列	123
	习题	124
<b>第6章</b>	<b>SQL 编程和存储过程</b>	<b>125</b>
6.1	SQL 编程	125
6.2	游标	134
6.3	存储过程	138
6.3.1	创建存储过程	138
6.3.2	查看、删除和修改存储过程	142

*6.4	触发器 .....	145
6.4.1	触发器简介 .....	145
6.4.2	创建触发器 .....	145
6.4.3	查看、删除和修改触发器 .....	146
习题	.....	148
<b>第7章</b>	<b>关系数据库设计规范化</b> .....	<b>149</b>
7.1	问题的提出 .....	149
7.2	规范化 .....	150
7.2.1	函数依赖 .....	150
7.2.2	码 .....	152
7.2.3	范式 .....	152
7.3	数据依赖的公理系统 .....	155
7.4	模式的分解 .....	160
7.4.1	模式分解的原则 .....	160
7.4.2	分解的无损连接性和保持函数依赖性 .....	163
7.4.3	模式分解的算法 .....	164
习题	.....	166
<b>第8章</b>	<b>数据库设计</b> .....	<b>168</b>
8.1	数据库设计概述 .....	168
8.2	需求分析 .....	168
8.3	概念结构设计 .....	169
8.3.1	实体—联系模型 .....	169
8.3.2	设计过程 .....	173
8.4	逻辑结构设计 .....	175
8.5	数据库物理设计 .....	178
8.6	数据库的实施和维护 .....	179
习题	.....	180
<b>第9章</b>	<b>数据库安全性</b> .....	<b>181</b>
9.1	数据库安全性概述 .....	181
9.2	登录账户 .....	181
9.3	数据库用户 .....	184
9.4	角色 .....	186
习题	.....	191
<b>第10章</b>	<b>数据库应用</b> .....	<b>192</b>
10.1	数据库访问技术 .....	192
10.2	Visual C++ 连接 SQL Server 2000 .....	195
10.3	Java 连接 SQL Server 2000 .....	202
习题	.....	207
<b>参考文献</b>	.....	<b>209</b>

# 第 1 章 数据库概述

## 本章要求:

(1) 了解数据库的基本概念,了解数据库管理系统的作用及不同的数据库管理系统的产品,了解数据库系统的范围。

(2) 理解数据库系统的特点,在后续的章节中可以加深对这些特点的理解。

(3) 了解数据模型的作用,理解概念数据模型尤其是实体—联系模型表示现实世界的方式,理解逻辑数据模型尤其是关系模型的概念,了解关系、属性、元组、关系模式等概念,为第 2 章打下基础。

(4) 了解数据库外部的系统结构及各种结构的特点。

(5) 了解数据库内部的系统结构,理解两层映像及两个独立性,加深对数据库系统的理解。

## 1.1 数据库系统概述

数据库系统是随着计算机技术的不断发展,为了实现对数据统一有效的管理而出现的,自 20 世纪 60 年代中期以来,数据库系统经历了三个发展阶段:第一阶段为层次与网状数据库系统,主要支持层次与网状模型;第二阶段为关系数据库系统,主要支持关系模型。20 世纪 80 年代以来,随着计算机硬件技术的不断提高和计算机应用的普及,产生了很多新的应用领域,这也给数据库系统提出了很多新的要求,由此产生了很多新型数据库,如面向对象数据库、分布式数据库等。由此,进入了数据库系统的第三个发展阶段。目前,关系数据库系统仍然占据着数据库应用的主流,所以在本章中的重点介绍内容为关系数据库。

### 1.1.1 数据库基本概念

数据库(DataBase,简称 DB)是相互关联的数据的集合。这里数据的形式可以是多种多样的,可以是文字、数字、图形、视频、声音等,如学生的信息、股票市场的数据、各个城市的地图、电视剧等。只有逻辑上相关的数据集合才可以定义为数据库,因此数据库是一个企业、组织或机构中需要保存和处理的所有数据。如,学校数据库可以包括学生的信息、教师的信息、课程的信息及学生选课、课程安排等信息。

一个数据库可以为多个用户和多个应用服务,从而实现数据的共享。如,学校数据库可以供学生信息管理系统、就业管理系统、排课系统、学生成绩管理系统等多个应用使用,使用该数据库的用户可以包括辅导员、教师、学生等。

数据库管理系统(DataBase Management System,简称 DBMS)是对数据库进行管理的通用的软件系统,位于用户和操作系统之间,可以对数据库的建立和维护、数据库的运行、



数据库的存取等实现方便有效的管理。目前,常用的关系数据库管理系统包括 Oracle 公司的 Oracle、Microsoft 公司的 SQL Server、IBM 公司的 DB2、原属 MySQL 公司的开源数据库 MySQL 等。

数据库、数据库管理系统加上应用程序、数据库管理员和用户构成了数据库系统。数据库系统的组成如图 1-1 所示。

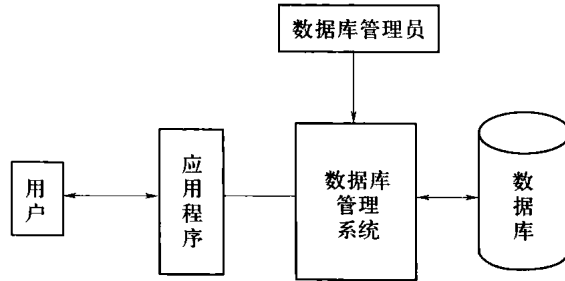


图 1-1 数据库系统的组成

程序设计人员可以利用各种开发工具(如 VC++、JAVA 等)开发以数据库为基础的面向某个具体应用的应用程序。应用程序通过数据库管理系统对数据库进行操作,完成数据的存取等各类操作。

数据库管理员的职责是对数据库的运行进行监督和管理,保证数据库的正常运行。

### 1.1.2 数据库系统的特点

数据库系统是在文件系统的基础上发展起来的,与文件系统相比,数据库系统有自身的很多优势,主要有以下几个方面:

#### 1. 采用数据模型表示数据,数据实现结构化

数据库系统采用数据模型表示数据,从而隐藏了数据的存储结构和存取方法等细节。目前,有很多种数据模型,不同的数据库系统采用不同的数据模型,对关系数据库而言,采用的是关系模型,图 1-2 是一个关系模型的实例。可以看出,关系数据库系统为用户提供了关系的名称及各个数据项的名称,例如,要得到学生的学号和姓名,只需要一条 SQL 语句:

```
SELECT 学号,姓名
FROM student
```

用户不必了解每个数据项的长度、存储的位置及存取方法等细节,因此对用户来说更容易理解和接受。

学号	姓名	性别	年龄	所在系
070811111	张明	女	20	信息
070811117	宋超	男	21	信息
070817001	王敏	女	19	外语
070817002	李娜	女	20	外语

图 1-2 student 关系

## 2. 支持数据共享,控制数据冗余

对数据库系统来说,一个数据库可以被多个用户或多个应用程序访问,因此数据库系统支持数据共享,不必为每个用户或应用程序建立自己的数据库,控制了数据的冗余。数据库系统提供了视图机制,使得不同的用户和应用可以创建和使用自己的视图。图 1-3 是针对 student 和 sc 关系的两个不同的视图。

学号	课程号	成绩	学号	课程号	成绩
070811111	080110B	100	070817001	080110B	90
070811111	080602A	90	070817001	080602A	90

学号	姓名	总成绩
070811111	张明	190
070817001	王敏	180

系名	平均年龄
信息	20.5
外语	19.5

图 1-3 sc 关系、学生总成绩视图和每个系学生的平均年龄视图

数据库系统具有并发控制机制,使得当多用户或多个应用程序同时访问数据库时结果是正确的。例如,多个飞机售票点同时售票时保证一个座位只能分配给一个乘客,并且数据库中的数据是正确的。

## 3. 数据独立性高

数据库系统有两种数据独立性:物理独立性和逻辑独立性。物理独立性是指当数据库的物理存储结构发生改变时,数据库的逻辑结构可以不变。逻辑独立性是指当数据库的逻辑结构发生改变时,应用程序不需要修改。数据独立性实际上是把数据定义与应用程序分离,对数据的定义和操作由数据库管理系统来完成,从而不必在应用程序中定义数据,减少了应用程序的工作量。

## 4. 数据由 DBMS 统一管理和控制

数据库管理系统是对数据库进行管理的软件,通常包括以下功能:

(1) 数据的完整性控制。数据的完整性是指数据库中数据的正确性和相容性。数据库管理系统需要提供定义完整性约束条件的机制、对完整性的检查机制及违约处理机制。例如,在 SQL Server 2000 中提供了主码约束、外码约束、唯一约束、核查约束、规则、默认值等机制,当用户修改数据库中的数据时,系统首先检查该操作是否满足完整性约束条件,如果不能,则拒绝执行该操作,从而保证数据库数据的完整性。

(2) 数据的安全性控制。数据的安全性是保护数据库中的数据,防止遭到非法的访问或破坏。在 SQL Server 2000 中提供了在操作系统、服务器、数据库、对象这四个级别的安全性来保护数据库中的数据。当用户登录数据库系统时,SQL Server 可以核对用户名和密码,禁止非法用户进入系统。同时,系统的合法用户在进行存取数据库等操作时,也要验证该用户是否有执行这个操作的权限,防止对数据库的非法存取,保证数据的安全。

(3) 数据库备份和恢复。数据库系统在运行过程中,不可避免地会发生各种各样的故障,如硬件故障、软件故障、操作失误及恶意破坏等,因此数据库管理系统要能够提供对数据库备份和恢复的机制,使得在数据库发生故障时,数据库管理系统能够根据数据库的备份将数据库恢复到某一个已知的正确状态。

(4) 并发控制机制。数据库管理系统能够提供并发控制机制以保证多用户或多应用同时访问数据库中的数据时,数据仍然是正确的。并发控制机制主要通过事务处理来实现,将一些操作序列定义为事务,事务是不可分割的,要么全部执行,要么不执行,从而避免事务在执行过程中受到并发事务的干扰产生错误的结果。

## 1.2 数据模型

数据模型是数据特征的抽象,用来对数据库中的数据进行描述。在数据模型中,需要精确地描述数据、数据之间的联系及数据的完整性约束。目前存在很多数据模型,可以将这些模型划分为两类,它们分属于不同的层次:概念数据模型和逻辑数据模型,见图 1-4。

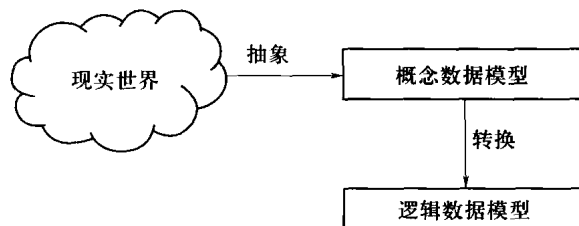


图 1-4 现实世界的抽象过程

### 1.2.1 概念数据模型

概念数据模型是对现实世界的第一层抽象,与具体的 DBMS 无关。在概念数据模型中常用的是实体—联系模型(E-R 模型),实体—联系模型包括实体、属性和联系等概念。实体是现实世界中各种事物的抽象,如学生、教师、部门等都是实体。属性是实体的特征或性质,如学生实体的属性有:学号、姓名、性别、年龄等。具有相同属性的实体集合称为实体型,由实体名和一组属性来定义。如学生实体型为:学生(学号,姓名,性别,年龄)。联系是实体之间的联系,可以分为一对一联系、一对多联系和多对多联系等。

实体—联系模型可以用实体—联系图(E-R 图)表示。图 1-5 是一个实体联系图的实例,表示售货员、商品和顾客三个实体的属性及实体间的联系。其中,矩形表示实体型,椭圆形表示实体的属性,菱形表示实体间联系。在实体联系图中,实体和联系连线上的字母表示联系的类型。关于实体—联系模型将在第 8 章详细论述。

实体—联系模型独立于具体的 DBMS,是各种逻辑数据模型的基础,只有转换为某个逻辑数据模型才能在 DBMS 中实现。

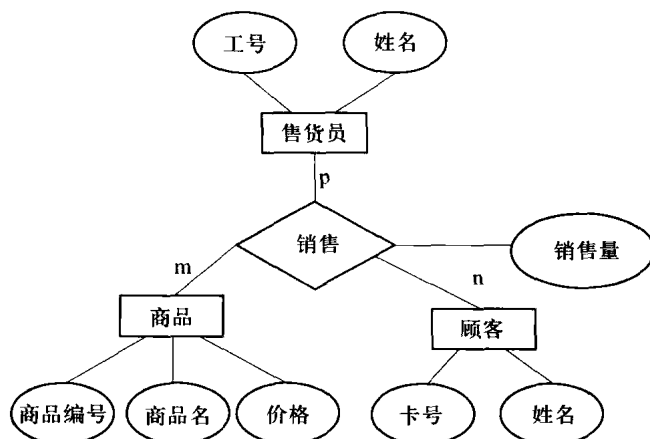


图 1-5 E-R 图实例

### 1.2.2 逻辑数据模型

逻辑数据模型是具体的 DBMS 所支持的数据模型,包括关系数据模型、层次数据模型、网络数据模型等,由于关系数据模型被大部分 DBMS 所采用,因此主要介绍关系数据模型。

图 1-6 是一个关系的实例。

课程号	课程名	学分
080110B	数据库原理与应用	4
080602A	软件工程	3

图 1-6 课程关系

关系数据模型的数据结构非常简单,如图 1-6 所示的关系,即二维表。

表中的每一列称为关系的属性,有相应的属性名。表中的每一行称为一个元组,因此关系可以看作元组的集合,每个元组代表一个实体,具有一定的含义。元组中的一个属性值称为分量。

对关系的描述称为关系模式,一般表示为:

关系名(属性 1,属性 2,……,属性  $n$ )

如上面的课程关系可以表示为:

课程(课程号,课程名,学分)

在关系数据模型中,实体间的联系也是用关系来表示。例如,图 1-7 的关系表示图 1-5 的销售联系。

关系数据模型的操作主要包括查询和更新数据两大类,操作的对象和操作结果都是关系,即元组的集合,同时关系数据模型的存取路径对用户透明,用户只需指出“做什么”,而不用考虑“怎么做”的问题,从而简化了操作的过程,提高了数据的独立性。例如,如果要查询卡号为 00100010 的顾客的购买记录,在 SQL Server 中只需要通过下面的 SQL 语句来完成:

售货员工号	商品编号	顾客卡号	销售量
S0001	121290	00100010	1
S0002	569878	00100099	3

图 1-7 销售联系对应的关系

```
SELECT 售货员工号,商品编号,顾客卡号,销售量
FROM 销售
WHERE 顾客卡号 = '00100010'
```

查询的结果仍然是元组的集合,即关系,如图 1-8 所示。

售货员工号	商品编号	顾客卡号	销售量
S0001	121290	00100010	1

图 1-8 查询结果

关系数据模型的完整性约束条件包括实体完整性、参照完整性和用户定义的完整性。其中,实体完整性是指关系的每个元组都有唯一标识。例如,课程号可作为课程关系的唯一标识,给定一个课程号,在课程关系中可以找到唯一的一个元组。

参照完整性是指多个表之间的关系,例如,图 1-9 是 SQL Server 的一个关系图,表示了学生、课程和选课之间的关系。选课关系中的学号取值必须是学生关系中某个学号值,选课关系中的课程号取值必须是课程关系中某个课程号值,这样保证选课关系表示的学生和课程都是实际存在的。

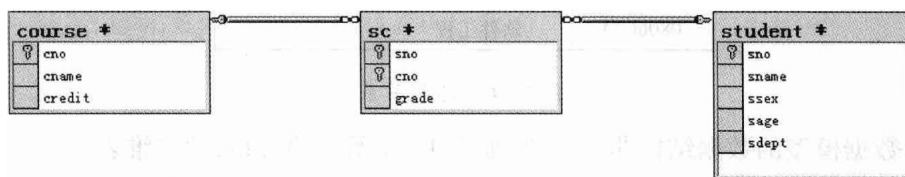


图 1-9 关系图

关系数据模型中,表以文件形式存储。在 SQL Server 中,将表存储在数据文件中,存储形式是由 SQL Server 完成的,一个表可以存储在多个数据文件中。

关系数据模型可由实体—联系模型自动转换,实体—联系模型中的实体和联系都可以转换为关系,而实体的属性可以转换为关系的属性,具体的转换规则将在第 8 章讲述。

### 1.3 数据库系统结构

数据库系统结构可以分为数据库内部的系统结构和数据库外部的系统结构。从数据库最终用户的角度来看,数据库外部的系统结构可以分为单用户结构、主从式结构、分布式结构、客户机/服务器结构和浏览器/服务器结构等。从数据库管理系统的角度看,数据库内部的系统结构采用三级模式结构。下面将分别介绍。

### 1.3.1 数据库外部的系统结构

数据库外部的系统结构可以分为以下几类：

1. 单用户结构:这是早期的数据库系统,系统的体系结构如图 1-10 所示。在这种结构中,数据库系统(应用程序、DBMS 和数据)装在一台计算机上,供一个用户使用,不同机器之间不能共享数据,数据冗余度大。

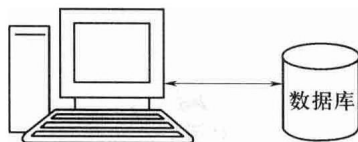


图 1-10 单用户结构

2. 主从式结构:主从式结构是对单用户结构的扩展,系统的体系结构如图 1-11 所示。在这种结构中,一台主机连接多个终端,应用程序、DBMS 和数据都装在主机的上,所有的处理任务由主机完成。多个终端可以并发的存取数据库,共享主机的数据。这种体系结构简单,易于维护,但是随着终端数量的增加,主机的任务会越来越繁重,从而降低系统的性能。一旦主机出现故障,整个系统都无法使用,系统的可靠性依赖主机。

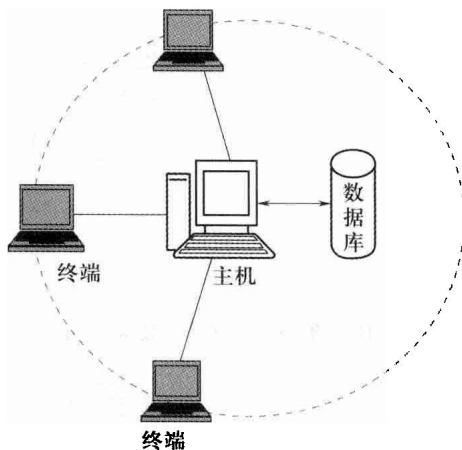


图 1-11 主从式结构

3. 分布式结构:分布式结构是计算机网络发展的产物,系统的体系结构如图 1-12 所示。数据库中的数据在逻辑上是一个整体,但物理的分布在网络中的不同结点上,每个结点上的主机又可以连接多个终端。网络中的每个结点都可以独立地处理本地数据,也可以存取异地数据。这种结构适合于地理上分散的公司和团体对数据处理的要求。但是数据的分布存放给数据的管理和处理带来困难,并且存取异地数据会受到网络传输速度的影响。

4. 客户机/服务器结构:客户机/服务器结构把 DBMS 和应用分开,将网络中某个结点专门执行 DBMS 功能,称为数据库服务器。网络中的其他结点安装开发工具和应用程序,称为客户机。这种结构可以分为集中式结构和分布式结构。集中式结构只有一台数据库服务器,体系结构如图 1-13 所示。分布式结构有多台数据库服务器,体系结构如图 1-14 所示。

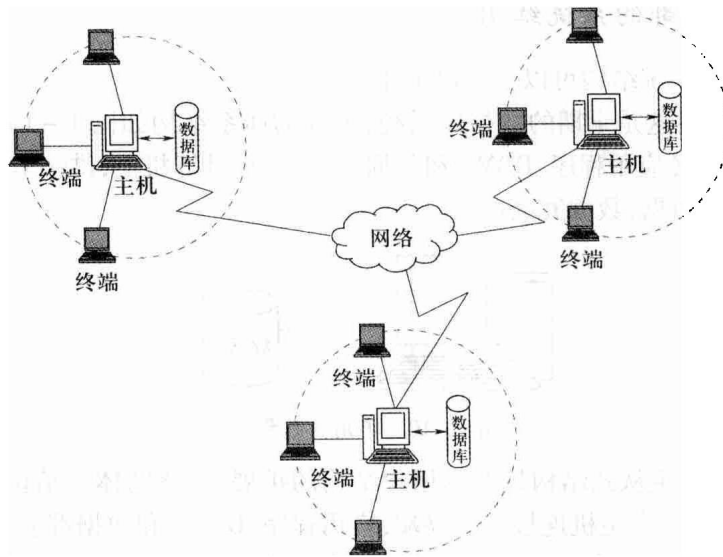


图 1-12 分布式结构

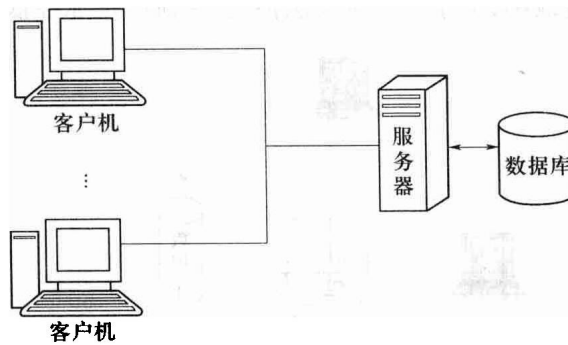


图 1-13 集中式客户机/服务器结构

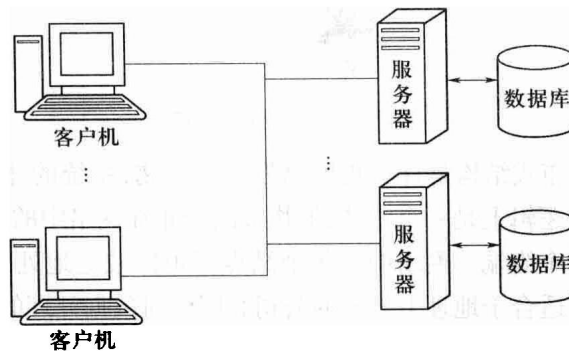


图 1-14 分布式客户机/服务器结构

客户机的请求被传送到数据库服务器,服务器处理完毕后,将结果返回给客户机。这种结构的优点在于客户机和服务器可以在不同的硬件和软件平台上运行,客户机可以使用不同的开发工具和应用程序。缺点在于需要为每一台客户机安装开发工具和应用程序,维护困难,工作量大,浪费系统资源。

5. 浏览器/服务器结构:这种结构的体系结构如图 1 - 15 所示。客户机端是统一的浏览器界面,用户容易掌握,减少了维护的工作量。服务器端分为 Web 服务器和数据库服务器,Web 服务器解析用户的数据处理请求,将对数据库的处理工作交给数据库服务器完成,数据库服务器将处理的结果返回给 Web 服务器,Web 服务器将结果以网页的形式显示给用户。这种结构能够支持成千甚至上万的用户,维护成本低,是应用最为广泛的数据库体系结构。

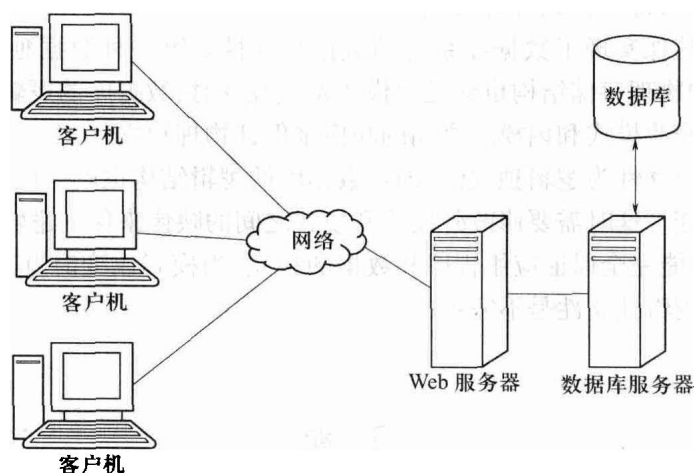


图 1 - 15 浏览器/服务器结构

### 1.3.2 数据库内部的系统结构

数据库系统提供了三种数据库模式:外模式、模式和内模式。三种模式的关系如图 1 - 16所示。

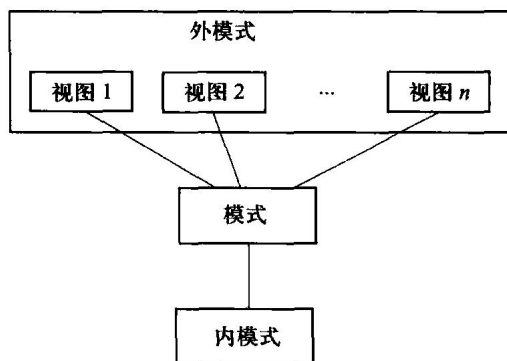


图 1 - 16 数据库的三级模式结构

模式是数据库中全体数据的逻辑结构和特征的描述,是数据库系统结构的中间层,因此它既不涉及数据的物理存储结构,也与具体的应用程序无关。模式通常以某种数据模型来描述,一个数据库只有一个模式。定义模式时不仅要定义数据本身,而且要定义数据之间的联系。

外模式是数据库中局部数据的逻辑结构和特征的描述,是与某一应用有关的数据的逻辑表示。一个数据库有多个外模式,不同的外模式可以为不同的应用程序服务,但一个



应用程序只能使用一个外模式。在关系数据库系统中,可以在表的基础上建立多个视图,如前面的学生总成绩视图和每个系学生的平均年龄视图,视图提供了以多种角度看待数据的机制,视图的集合构成一个数据库的外模式。

内模式是数据物理结构和存储方式的描述,一个数据库只有一个内模式。数据库的内模式包括存储策略和存取路径的描述。

由于数据库系统的三级模式结构,数据库管理系统提供了两种映像:

外模式/模式映像和模式/内模式映像

正是这两种映像实现了数据库系统的数据独立性。第一种数据独立性为物理独立性,即当数据库的物理存储结构也就是内模式发生改变时,数据库的逻辑结构也就是模式不变。这时需要修改模式和内模式之间的映像来保证物理独立性。

第二种数据独立性为逻辑独立性,即当数据库的逻辑结构也就是模式发生改变时,数据库的外模式不变。这时需要修改外模式和模式之间的映像来保证逻辑独立性。需要注意,逻辑独立性不能完全保证应用程序和数据的独立,当模式改变时,应用程序可能仍然需要修改。因此逻辑独立性是不完备的。

## 习 题

1. 简述数据库系统的特点。
2. 常见的数据库管理系统的产品有哪些?它们的特点是什么?
3. 什么是关系?什么是关系模式?请举例说明。
4. 关系模型的特点有哪些?
5. 说明数据库系统的三级模式结构的含义。
6. 说明数据库系统的两层映像及与独立性的对应关系。
7. 数据库系统外部的体系结构有哪几种?常用的有哪几种?请结合实际举例说明。
8. 数据库管理系统的功能有哪些?