

数据库

系统原理

编著 赵莉萍

shujuku xitong yuanli

华东理工大学出版社

数据库系统原理

赵莉萍 编著

华东理工大学出版社

图书在版编目(CIP)数据

数据库系统原理/赵莉萍 编著. —上海:华东理工大学出版社, 2002. 8

ISBN 7 - 5628 - 1273 - X

I. 数... II. 赵... III. 数据库系统 - 高等学校 - 教材 IV. TP311.13

中国版本图书馆 CIP 数据核字(2002)第 043544 号

数据库系统原理

赵莉萍 编著

出版	华东理工大学出版社	开本	787 × 1092 1/16
社址	上海市梅陇路 130 号	印张	15
邮编	200237 电话 (021)64250306	字数	362 千字
网址	www.hdlgpress.com.cn	版次	2002 年 8 月第 1 版
经销	新华书店上海发行所	印次	2002 年 8 月第 1 次
印刷	上海长阳印刷厂	印数	1 - 2050 册
ISBN 7 - 5628 - 1273 - X/TP•114		定价:25.00 元	

内 容 提 要

本书主要包括:数据库系统概述部分、关系数据库部分、关系数据库标准语言 SQL 部分、关系数据库设计理论部分、数据库保护部分等。以 SQL2 为标准详细介绍了 DDL、QL、DML 三大语言,并列举了大量的实例。同时对当代流行的关系数据库的过程化语言中的存储过程、存储函数、触发器等作了介绍。关系数据库实例 ORACLE 8i 是本书的一大特点,结合一个关系数据库,使学生进一步掌握数据库的内部机理,使抽象的理论转化为直观的演示。

本书适合作高等院校的专业课教材,也适合作工程技术人员的自学用书。

前 言

为了数据库技术的高速发展,需编撰迎合当代数据库技术发展趋势的数据库专业教材,以满足高等院校计算机及其相关专业学生对当今流行数据库技术专业知识的渴求。本教材结合当代流行的数据库,例如 ORACLE、SYBASE、SQL * Server 等,对数据库技术及其内部机制作了深入浅出的阐述。

本教材的主要内容包括:概述部分对数据库及其管理系统基本概念作了简介,其中包含数据模型,数据库的体系结构,面向对象的数据库,数据字典等内容的介绍;关系数据库部分对关系的由来,关系模式的描述,关系代数及查询优化作了详细的阐述;标准化 SQL 语言部分以 SQL2 为标准详细介绍了 DDL、QL、DML 三大语言,并列举了大量的实例,同时对当代流行的关系数据库的过程化语言中的存储过程、存储函数、触发器等作了介绍;关系数据库设计理论部分包含范式、Armstrong 公理、关系模式的分解算法,并列举相关的实例;数据库设计部分对数据库设计的全过程作了详细的描述;数据库保护部分以 ORACLE 数据库为实例,针对数据库的安全性、完整性、并发性和数据恢复作了详细的描述;数据库管理系统的实现技术是本教材的一个特点,针对开发一个 DBMS 所需的知识和技术,详细说明了数据库管理系统的内部机制及其实现技术;关系数据库实例 ORACLE 8i 是本书的又一大特点,结合一个关系数据库,使学生进一步掌握数据库的内部机理,使抽象的理论转化为直观的演示;数据库的发展技术部分可作为读研究生的学生继续深造的参考章节。

本教材在编写过程中力求由浅入深、循序渐进,对基本概念讲述清楚,并有大量的实例,具有理论联系实践的特点,书中插有许多图片,内容丰富,不仅适合作教材,而且适合用作工程技术人员的自学用书。

本教材另有配套的基于 ORACLE 软件平台的实验讲义,结合上机实验,才能收到良好的学习效果。

由于作者水平有限,书中难免存在一些缺点和错误,殷切希望广大读者批评指正。

编 者

2002 年 8 月

目 录

1 数据库系统引论	(1)
1.1 引言	(1)
1.1.1 数据、数据处理、数据库与数据库管理系统	(1)
1.1.2 数据库技术的产生与发展	(2)
1.1.3 数据库系统的主要特征	(4)
1.2 数据描述	(7)
1.2.1 现实世界	(7)
1.2.2 信息世界	(7)
1.2.3 计算机世界	(7)
1.2.4 常用术语的说明	(8)
1.3 数据模型	(9)
1.3.1 数据模型的要素	(9)
1.3.2 层次模型	(10)
1.3.3 网状模型	(10)
1.3.4 关系模型	(11)
1.3.5 E-R 数据模型	(13)
1.3.6 面向对象数据模型	(16)
1.4 数据库系统结构	(21)
1.4.1 数据库系统的模式结构	(21)
1.4.2 数据库系统的体系结构	(23)
1.5 数据库管理系统	(25)
1.5.1 数据库管理系统的功能	(25)
1.5.2 数据库管理系统的组成	(26)
1.6 数据库语言	(27)
1.6.1 数据描述语言	(27)
1.6.2 数据操纵语言	(29)
1.7 数据库管理员	(30)
1.8 数据字典	(30)
1.8.1 数据字典的基本概念	(30)
1.8.2 DBMS 和数据字典	(31)
1.8.3 数据字典的任务	(31)
1.8.4 数据字典的用途	(32)
1.9 数据库系统的工作流程	(32)
习题 1	(33)
2 关系数据库	(34)

2.1	关系数据结构概述	(34)
2.1.1	关系数据结构	(34)
2.1.2	关系操作	(34)
2.1.3	完整性约束	(35)
2.2	关系数据结构	(35)
2.2.1	关系	(35)
2.2.2	关系模式	(36)
2.2.3	关系数据库	(37)
2.3	关系的完整性	(37)
2.3.1	实体完整性	(37)
2.3.2	参照完整性	(38)
2.3.3	用户定义完整性	(38)
2.4	关系代数	(39)
2.4.1	传统的集合运算	(39)
2.4.2	专门的关系运算	(40)
2.4.3	扩充的关系操作	(43)
2.5	查询优化	(44)
2.5.1	关系代数表达式的优化	(44)
2.5.2	关系代数表达式的等价变换规则	(45)
2.5.3	优化的一般策略	(47)
2.5.4	关系代数的优化算法	(47)
习题 2		(50)
3	关系数据库标准语言 SQL	(51)
3.1	SQL 概述	(51)
3.1.1	SQL 的产生与发展	(51)
3.1.2	SQL 数据库的体系结构	(51)
3.1.3	SQL 的组成	(52)
3.2	SQL 的数据定义	(52)
3.2.1	模式的定义和撤消	(52)
3.2.2	基本表的定义、修改和撤消	(53)
3.2.3	索引的定义和撤消	(56)
3.3	SQL 的数据查询	(56)
3.3.1	单表查询	(57)
3.3.2	连接查询	(59)
3.3.3	嵌套查询	(61)
3.3.4	集合查询	(63)
3.4	SQL 的数据更新	(64)
3.4.1	数据插入	(64)

3.4.2 数据修改	(65)
3.4.3 数据删除	(66)
3.5 视图及其操作	(66)
3.5.1 定义视图	(66)
3.5.2 视图的撤消	(68)
3.5.3 视图的查询	(68)
3.5.4 视图的更新	(69)
3.5.5 视图的优点	(69)
3.6 T-SQL 和 Oracle 中的存储过程、函数和触发器(*)	(70)
3.6.1 T-SQL 中存储过程、函数和触发器	(70)
3.6.2 Oracle 中存储过程、函数和触发器	(73)
3.7 嵌入式的 SQL(*)	(75)
3.7.1 SQL 和宿主语言之间的通信接口	(76)
3.7.2 嵌入式 SQL 的使用技术	(78)
习题 3	(82)
 4 关系数据库设计理论	(84)
4.1 数据依赖	(84)
4.1.1 关系模式中的数据依赖	(84)
4.1.2 数据依赖的有关概念	(84)
4.1.3 数据依赖对关系模式的影响	(86)
4.2 范式	(87)
4.2.1 第一范式	(87)
4.2.2 第二范式	(88)
4.2.3 第三范式	(89)
4.2.4 BC 范式	(90)
4.2.5 多值依赖与第四范式	(90)
4.3 函数依赖的公理系统	(92)
4.3.1 Armstrong 公理	(92)
4.3.2 Armstrong 公理的推论	(93)
4.3.3 属性的闭包	(93)
4.3.4 最小函数依赖集	(94)
4.4 关系模式的规范化	(96)
4.4.1 关系模式的规范化步骤	(96)
4.4.2 关系模式的分解	(97)
4.4.3 模式的分解算法	(100)
4.4.4 多值依赖公理及其分解算法	(102)
习题 4	(103)

5 数据库设计	(105)
5.1 数据库设计概述	(105)
5.1.1 数据库设计的内容和特点	(105)
5.1.2 数据库设计方法简介	(106)
5.2 需求分析	(108)
5.3 概念模型设计	(110)
5.3.1 设计局部的 E-R 模型	(111)
5.3.2 将各局部的 E-R 模型综合成总体的 E-R 模型	(112)
5.4 逻辑结构设计	(115)
5.4.1 概念模型向网状模型转换	(115)
5.4.2 从 E-R 图向关系模型转换	(117)
5.4.3 规范化理论的应用	(119)
5.5 数据库物理设计	(120)
5.5.1 数据簇集设计	(121)
5.5.2 索引的选择	(123)
5.6 数据库的实施与维护	(126)
5.6.1 数据库数据的装入	(126)
5.6.2 数据库的试运行	(127)
5.6.3 数据库的运行和维护	(127)
习题 5	(128)
 6 数据库保护	 (130)
6.1 数据库的完整性	(130)
6.1.1 实体完整性	(130)
6.1.2 参照完整性	(130)
6.1.3 用户定义完整性	(131)
6.1.4 对约束的命名、撤消和添加操作	(131)
6.1.5 约束的使能与撤消以及延迟与撤消	(132)
6.1.6 触发器	(132)
6.2 数据库的安全性	(133)
6.2.1 账户的安全性	(134)
6.2.2 用户对数据库对象的存取权限	(134)
6.2.3 DBA 对系统级的权限	(135)
6.3 数据的并发控制	(135)
6.3.1 事务	(135)
6.3.2 并发操作调度	(137)
6.3.3 并发事务的可串行化	(138)
6.3.4 并发事务的可恢复性	(141)
6.3.5 事务的并发操作带来的问题	(142)

6.3.6 数据封锁技术	(143)
6.3.7 基于时标的并发控制	(149)
6.4 数据库的恢复	(150)
6.4.1 数据库系统遭遇的故障	(150)
6.4.2 恢复和原子性的联系	(151)
6.4.3 基于日志的恢复技术	(151)
习题 6	(155)
7 数据库管理系统及其实现技术	(157)
7.1 数据库管理系统概述	(157)
7.1.1 数据库管理系统结构简介	(157)
7.1.2 数据库管理系统的目标	(158)
7.1.3 数据库管理系统的基本功能	(159)
7.1.4 数据库管理系统与操作系统的关系	(159)
7.2 数据库管理系统实现	(161)
7.2.1 DBMS 程序模块的组成	(161)
7.2.2 DBMS 的进程结构	(163)
7.2.3 DBMS 的系统结构	(165)
7.3 数据库语言的处理	(168)
7.3.1 语言翻译处理层的任务和工作步骤	(168)
7.3.2 解释方法	(169)
7.3.3 预编译方法	(170)
7.4 数据存取层	(171)
7.4.1 存取层的系统结构	(172)
7.4.2 存取层的功能子系统	(173)
7.5 缓冲区管理	(175)
7.6 数据库物理组织	(175)
7.6.1 数据库物理组织概述	(175)
7.6.2 数据库实现联系的方法	(178)
习题 7	(179)
8 关系数据库 ORACLE 8i	(181)
8.1 Oracle 的体系结构	(181)
8.1.1 数据库	(181)
8.1.2 数据库实例	(182)
8.1.3 数据库实例的启动与关闭	(183)
8.1.4 数据库存储	(184)
8.2 数据库物理文件的维护	(186)
8.2.1 控制文件的维护	(186)

8.2.2 重做日志文件的维护	(187)
8.2.3 数据文件的维护	(189)
8.3 Oracle 数据库基本对象的管理	(190)
8.3.1 段的类型	(190)
8.3.2 基本表	(191)
8.3.3 索引	(192)
习题 8	(195)
 9 数据库系统的研究与发展	 (196)
9.1 分布式数据库系统	(196)
9.1.1 分布式数据库系统概述	(196)
9.1.2 分布式数据库系统组成	(200)
9.2 客户/服务器结构的数据库系统	(206)
9.2.1 概述	(206)
9.2.2 客户/服务器结构数据库系统	(208)
9.2.3 客户/服务器结构数据库系统实现技术	(212)
9.2.4 客户/服务器结构数据库系统的优点	(216)
9.3 面向对象的数据库系统	(217)
9.3.1 传统的关系数据库系统的不足之处	(218)
9.3.2 “对象”概念的实质	(218)
9.3.3 面向对象数据库技术的产生	(219)
9.3.4 面向对象模型	(219)
9.3.5 面向对象的数据库管理系统	(220)
9.4 一些新型的数据库系统	(220)
9.4.1 演绎数据库	(220)
9.4.2 模糊数据库	(222)
9.4.3 模糊演绎数据库	(223)
9.4.4 巨型数据库系统	(224)
9.4.5 主动数据库	(224)
9.4.6 多媒体数据库	(225)
9.4.7 集成的工程数据库	(226)
9.4.8 联邦数据库	(226)
9.4.9 时态数据库	(226)
习题 9	(227)

1 数据库系统引论

1.1 引言

数据库技术是计算机软件领域的一个重要分支之一,目前已形成相当规模的理论体系和实用技术。在计算机的三大应用(科学计算、数据处理和过程控制)中,数据处理约占70%的比重,而数据库技术就是作为一门数据处理技术发展起来的,尤其在信息变为经济发展重要资源的时代,数据库的建设规模和使用水平已经成为衡量一个国家信息化程度的重要标志。本章介绍数据库中的主要概念,使读者对数据库系统概貌有所了解。

1.1.1 数据、数据处理、数据库与数据库管理系统

1.1.1.1 数据

我们在日常生活中接触到大量数据,如银行账目数据、学校的教学管理数据、企业的生产管理和产品销售数据等等。这些数据都是人们生活中所离不开的,它们可以被视为人类社会中一种极其重要的资源。

然而这些数据之所以有价值,就是因为这些数据是表现信息、载荷信息的物理符号。简单地说,信息是向人们提供关于现实世界各种事实,反映了客观事物的物理状态。例如,某一国家的汽车产量是30万辆,这个30万辆是一个数据,而它又表示了这个国家生产汽车的能力这一信息。由此可见,信息是用数据来表示的,但两者是不可分离而又有一些区别的概念。一方面并非任何数据都能表达信息,信息是数据的内涵,是对数据的语义解释;另一方面,数据是信息的符号表示或载体,是信息的具体表现。

1.1.1.2 数据处理

对数据进行综合推导,得出新的数据,这些新的数据则表达了新的信息。例如,在商品生产中,生产管理者根据某种商品历年销售数量及最近的市场需求调查,获得了许多数据,再对这些数据进行加工,就得出了有关商品的市场预测信息。生产管理者就可以根据这些信息进行分析和评价,作出对该产品是增产、减产还是停产的决策,并指挥生产部门的生产活动。这种从收集数据,到加工成信息,进行评价和决策,再指挥实践活动,从而产生新的数据的循环过程称为信息循环。

诸如此类围绕信息所做的一系列工作,称为信息处理。一般地说,信息处理就是指信息的收集整理、加工、存储和传播等一系列活动的总和。因为信息是用数据表示的,所以对信息的处理又具体地体现在对数据的处理上。通过对原始数据(即尚未作评价的各种信息)的处理,产生新的数据(新的信息),作为结果。这一处理,包括对数据的收集、记录、分类、排序、存储、计算/加工、传输、制表和递交等处理,这就是数据处理的概念。经过处理的数据是精炼的数据,是能够反映事物或现象的本质和特征及其内在联系的数据。

数据处理的历史可以追溯到远古时代。原始人类的结绳记事,累计记数便是数据处理的雏形。随着社会生产和文明的日益发展,信息的概念就趋于越来越复杂和深化,信息已经支配着人类的整个社会活动,所以有人把现代社会称为信息社会。研究信息的形态、传输、处理和存储理论的信息科学也就应运而生。

1.1.1.3 数据库

收集并抽取一个应用所需的大量数据之后,应将其保存起来以供进一步加工处理和抽取信息。保存的方法很多,例如人工保存,存放在文件中,存放在数据库中。其中数据库是存放数据的最佳场所,其原因将在 1.1.2 节中介绍。

所谓数据库就是长期存在计算机内、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,并可作为各种用户所共享。

1.1.1.4 数据库管理系统

收集并抽取一个应用所需的大量数据之后,如何科学地组织这些数据并将其存储在数据库中,又如何高效地处理这些数据呢?完成这些任务是一个软件系统,即数据库管理系统,又称 DBMS。

DBMS 是位于用户与操作系统之间的一层数据管理软件(见图 1-1)。

数据库在建立、运行和维护时由数据库管理系统统一管理、统一控制。数据库管理系统使用户方便地定义数据和操纵数据,并能保证数据的安全性、完整性,保证多用户对数据的并发使用及发生故障时的系统恢复。

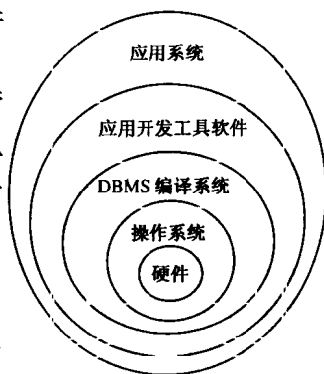


图 1-1 数据库在计算机系统中的地位

1.1.2 数据库技术的产生与发展

随着信息概念的深化和发展,数据处理也就得到了相应的发展,就整个数据处理的发展过程来看,大致可以分成三个阶段。

1.1.2.1 人工管理阶段

在 20 世纪 50 年代中期以前,计算机主要用于科学计算。当时的硬件状况是,外存只有纸带、磁带、卡片,没有磁盘等直接存取设备;软件状况是,没有操作系统,没有管理数据的软件,数据处理方式是批处理。这一时期数据与程序之间的关系可以用图 1-2 表示。

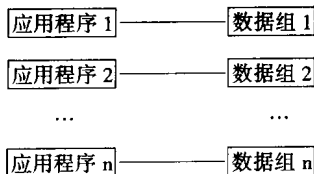


图 1-2 人工管理阶段数据和程序之间的关系

1.1.2.2 文件系统阶段

20 世纪 50 年代后期到 60 年代中期,计算机的应用范围逐渐扩大,计算机不仅用于科学计算,而且还大量用于管理。这时硬件已有了磁盘、磁鼓等直接存取设备;软件方面,操作系

统已经有了专门的数据管理软件,一般称为文件系统;处理方式不仅有了批处理,而且能够联机实时处理。由文件系统充当应用程序和数据文件之间的接口,这一时期应用程序和数据的关系如图 1-3 所示。

虽然这一阶段较人工管理阶段有了很大的改进,但仍存在很大的弱点,即文件本身还是基本上对应于一个或几个应用程序,或者说数据还是面向应用的。从逻辑结构上讲,还与前一阶段没有什么变化,尽管程序不必直接与文件打交道而有软件作为接口,但它仍然是一个不具有弹性的无结构的信息集合,存在着冗余度大、空间浪费、文件不易扩充等缺点,从而反映不了现实世界事物之间的广泛的内在联系。

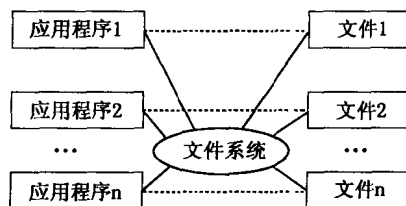


图 1-3 文件系统阶段数据和程序之间的关系

1.1.2.3 数据库系统阶段

20 世纪 60 年代后期以来,计算机用于管理的规模更为庞大,应用越来越广泛,数据量急剧增长,同时多种应用、多种语言互相覆盖地共享数据集合的要求越来越强烈。这时硬件已经有了大容量磁盘,硬件价格下降,软件价格上升,为编制和维护系统软件及应用程序所需的成本相对增加;在处理方式上,联机实时处理要求更多,并开始提出和考虑分布处理。在这种背景下,以文件系统作为数据管理的手段已经不能满足应用的要求,于是为解决多用户、多应用共享数据的需求,使数据能为尽可能多的应用服务,就出现了数据库技术,出现了统一管理数据的专门软件系统——数据库管理系统。

表 1-1 数据处理的三个阶段的比较

		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	无直接存取设备	磁盘、磁鼓	大容量磁盘
	软件背景	没有操作系统	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	数据管理者	人	文件系统	数据库管理系统
	数据面向对象	某个应用程序	某个应用程序	现实世界
	数据共享程度	无共享 冗余度大	共享性差 冗余度大	共享性高 冗余度小
	数据独立性	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 DBMS 提供数据安全性、完整性、并发控制和恢复

数据库技术克服了以前所有管理方式的缺点,试图提供一种完善的、更高级的数据管理方式。它的基本思想是解决多用户数据共享的问题,实现对数据集中统一管理,具有较高的数据独立性,并为数据提供各种保护措施。这一时期,数据库管理软件作为用户与数据的接

口,程序和数据的关系如图 1-4 所示。

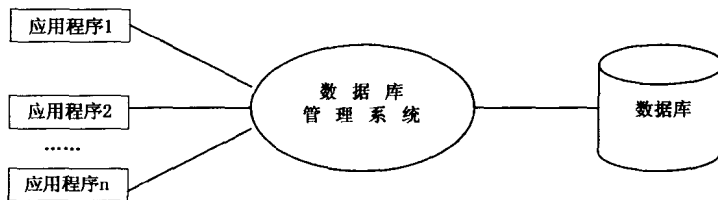


图 1-4 数据库阶段数据与程序之间的关系

三个阶段软硬件背景及其特点对比列于表 1-1。

1.1.3 数据库系统的主要特征

数据库技术之所以能够在近 20 年内如此快速的发展,受到计算机科学界普遍的重视,成为引人注目的一门新兴学科,是因为它具有独特的特征。它主要有如下的特征。

1.1.3.1 实现数据共享

这是促成数据库技术发展的一个重要原因,也正是数据库技术先进性的一个重要体现。数据的共享性包括三个方面。

(1) 当前的所有用户可以同时存取数据库的数据,这是十分清楚的。因为数据库的研制正是以当前用户为现实基础的。

(2) 未来的新用户与当前用户可以同时存取数据库中的数据。也就是说,数据库不仅可以为当前已知的多个用户服务,也可为将要使用该数据库的任何新用户服务,而且可以同时为多个新用户服务。

(3) 用户可以通过多种程序设计语言或命令语言与数据库接口。如 COBOL、C、FORTRAN、JAVA,以及终端命令和数据库管理系统提供的数据操纵语言等等,从而可以用多种语言使用数据库。

1.1.3.2 减少数据的冗余度

在数据库技术产生以前的文件系统中,用户根据各自的应用建立文件,即使不同的应用程序需要使用的数据中有许多是相同的,也必须建立各自的文件,其原因就在于文件系统中数据不能充分共享(文件系统中只提供文件一级共享,不提供记录级的共享)。因此使得存储的数据存在着大量的重复,这种现象称为数据的冗余。

下面我们通过一个例子来说明这个问题。

假设现在要管理的对象是一个学校,它由人事子系统、业务子系统、总务子系统组成,各子系统根据自己的特点及需要选择相应的数据格式,存放自己感兴趣的数据。它由这样三种记录组成,如图 1-5 所示。

人事记录

部	职工	姓	性	年	职	工	家	本	政	社
门	号	名	别	龄	务	资	庭	人	治	会
							出	成	历	关
							身	分	史	系

业务记录

部	职工	姓	性	年	职	工	学	专	外	教
门	号	名	别	龄	务	资	历	业	语	学
									程	历
									度	

总务记录

部	职	姓	性	年	职	工	参	家	高	住
门	工	名	别	龄	务	资	加	庭	校	房
号	号						工	人	教	面
							作	口	龄	积
							时			
							间			

图 1-5 学校的三个记录

在文件系统阶段,由这三个记录的集合组成三个文件:人事文件、业务文件、总务文件。这三个文件对应于三个不同的应用系统,它们之间互不联系。而在这三个文件中,有许多项目如部门、职工号、姓名、性别、年龄、职务、工资是相同的,而在文件系统中,数据与应用系统是一一对应的,所以必须分别存储,这样就势必有很大的数据冗余,浪费存储空间。而这种冗余还会带来其他问题,如在更新操作时,必须要进行多次更新操作,以更新某些冗余的副本,增加不必要的机器时间。再者由于多个副本的存在,系统容易出现不一致的信息。而在数据库中,是从整体观点来组织数据的,其数据不是面向个别的应用,而是为多种应用所共享的,从而避免了不必要的冗余。例如上面的例子在数据库中三个文件就不再是互不相干的了,而是将这三个文件视为统一的整体,也就是说,把这三个文件进行分解,重新组合,去掉重复部分,构成一个整体。图 1-6 给出经过重新组合后形成的数据结构。

从图中可以看出,我们设计了各部门都要用的职工记录,并带上了人事记录、业务记录和总务记录等,这样就把整个组织的数据结构化了。这就要求在描述数据时,不仅要描述数据本身,还要描述数据之间的联系,这种联系往往是通过存取路径来表现的。在这种数据结构化的情况下就大大地减少了数据的冗余度,并能节省空间、提高存取速度、避免不相容。

此外,由于是从整体的观点看数据,所以数据不是面向应用的,而是面向系统的。这样它的弹性大,可以扩充,应用数据时有灵活的方式,即可以取整体模型的各种合理的子集,用于不同的应用系统。

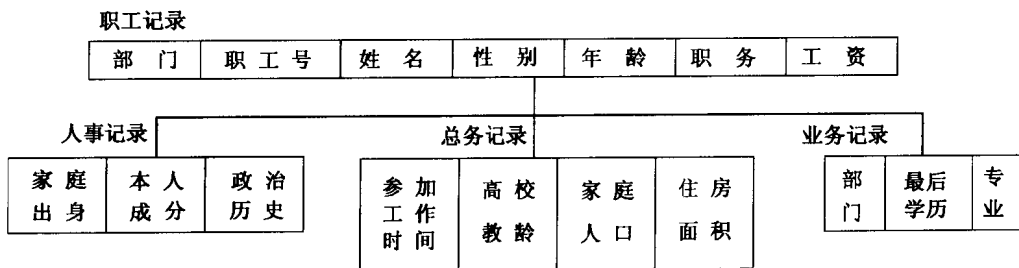


图 1-6 组合后的数据结构

从理论上说数据库中是没有冗余的,但实际上,许多数据库为了改善访问时间,或者由于采用比较简单的存取方法,数据库中还是存在着某种程度的冗余性。但是它去掉了有害的或不必要的冗余。

1.1.3.3 维护数据的独立性

所谓数据的独立性,就是应用程序不必因为数据的存储结构的变化而修改,即应用程序与数据的结构之间不存在依赖关系,这是数据库系统所努力追求实现的一个目标。数据库系统结构之所以复杂,这是一个重要原因。数据库系统的数据独立性可分为两级。

(1) 物理独立性。即数据库的物理结构的变化(如物理设备的更换、物理位置的变化、存取方法的改变等等)不影响数据库的逻辑结构,从而也就影响不到应用程序,不会导致应用程序的修改。

(2) 逻辑独立性。数据库总体逻辑结构的变化(如数据定义的修改,新数据类型的增加,数据间联系的变更等等)不致影响到数据的局部逻辑结构,由于用户原有应用程序是基于局部逻辑结构编写的,所以应用程序不必修改。

这两种独立性统称为数据独立性。数据独立性的目的,就是应用程序尽可能不受数据的影响。这两种数据独立性是靠数据库管理系统来实现的,从而大大减轻了程序员的负担。

1.1.3.4 加强对数据的保护

用户非常关心数据的安全性和可靠性。数据的安全可靠是一个数据库能否实用的关键问题。一般较完整的数据库系统对数据的保护有如下四个方面的内容。

1) 安全保密控制

主要是数据的保密性控制,即保护库中的数据以防止不合法的使用。一般可以采取如下三种方法,来保证数据的安全性:

(1) 将数据库中需要保密的部分与其他公用部分的数据隔离开来;

(2) 建立一些访问规则。如给用户以身份号,使某些特殊的数据只能由特定身份的用户使用;或给用户授予一定的权限去处理某些特定的数据,同时由系统检查是否有非法使用的情况发生,并采取相应的措施。对不同的操作,采用不同的保护级别,如有的用户只有查询权,有的用户则可以获得修改权等;

(3) 将数据以密码的形式存放于数据库内。

2) 完整性控制

数据的完整性包括数据的正确性、有效性和一致性。如一个数值型数据包含了诸如字母、特殊符号、字符,显然是不正确,是失去完整性的例子。此外,形式正确的数据也并非一定是有效的,例如月份必须是1~12的正整数,否则是无效的。表示同一事实的两个数据项之间的不一致性也是缺乏完整性的例子。数据库系统对数据的使用是集中控制的,因而有利于控制数据的完整性。

3) 并发控制

数据的不相容性主要是由于数据的共享而引起的。不同用户同时使用数据库,可能引起对数据的干扰。例如,当某个用户需要修改某一数据时,为此他必须先读出这个数据,修改后再写回去,如果在这个用户把被修改的数据写回数据库之前,别的用户再去读出这个正在被修改的数据就不合理了,就会引起错误。数据库管理系统的并发控制功能,将排除和避免这种错误的发生,保证数据的正确性。

4) 数据库恢复

数据库在运行过程中很难保证不受破坏,全局性的和局部性的破坏随时都可能发生。因为硬件和软件故障以及用户操作上的错误都是不可避免的。为了防止由于这些故障而破坏数据的正确性,数据库系统通过下述措施来保证数据的正确性:

(1) 定期将数据库中的内容拷贝到磁带上加以保存;

(2) 对每次应用数据库的过程进行记录,以便出错时检查错误的来源;

(3) 每当必须对数据库的内容进行修改时,将修改前的内容及修改后的内容记到磁带上保存起来,以备检查或重作时使用;

(4) 当发现对数据库的某个处理有错误时,消除有关处理的影响,以恢复正确的数据库。