

SHUJUKU YUANLI JI YINGYONG

数据库原理及应用

Shujuku Yuanli Ji Yingyong

主 编 何宗耀 吴孝丽

中国矿业大学出版社

China University of Mining and Technology Press

数据库原理及应用

主编 何宗耀 吴孝丽

中国矿业大学出版社

TP311.13-43
235

内 容 提 要

全书共分 10 章,全面系统地阐述了数据库技术的基本原理和应用,主要内容包括数据库系统概述、关系代数和关系运算、关系数据库系统、结构化查询语言 SQL、数据库的安全保护、数据库设计、T-SQL 程序设计和数据库系统的新发展。本书以 SQL Server 2008 为平台介绍数据库技术的实现,包括数据库的创建和数据表的维护、查询与统计、视图、存储过程和触发器的创建、用户管理和权限分配、约束和默认、数据库的备份和还原等内容,使读者能够通过 SQL Server 2008 平台深刻理解数据库技术的相关原理和技术,达到理论和实践紧密结合的目的。

本书内容循序渐进、深入浅出,概念规范、条理清晰,每章均有大量的代码和实例进行解释和说明,同时附有习题,从不同方面帮助读者了解和掌握相关知识点,从而确立完善的理论体系并提高实践能力。

本书可供大学本、专科计算机专业学生使用,亦可供相关工科专业学生及相关科研人员、工程技术人员学习和参阅。

图书在版编目(CIP)数据

数据库原理及应用/何宗耀,吴孝丽主编. —徐州:

中国矿业大学出版社,2014.9

ISBN 978 - 7 - 5646 - 2462 - 0

I. ①数… II. ①何… ②吴… III. ①数据库系统—
高等学校—教材 IV. ①TP311.13

中国版本图书馆 CIP 数据核字(2014)第 198210 号

书 名 数据库原理及应用

主 编 何宗耀 吴孝丽

责任编辑 潘俊成

出版发行 中国矿业大学出版社有限责任公司

(江苏省徐州市解放南路 邮编 221008)

营销热线 (0516)83885307 83884995

出版服务 (0516)83885767 83884920

网 址 <http://www.cumtp.com> E-mail: cumtpvip@cumtp.com

印 刷 徐州中矿大印发科技有限公司

开 本 787×1092 1/16 印张 18.75 字数 492 千字

版次印次 2014 年 9 月第 1 版 2014 年 9 月第 1 次印刷

定 价 33.00 元

(图书出现印装质量问题,本社负责调换)

前 言

云计算和大数据时代的到来,使“以数据处理为中心”的计算理念转换到“以数据为中心”,体现了数据为王;而数据库技术是数据处理的基本技术,同时也是计算机科学一个非常重要的部分,数据库技术以及数据库应用正以日新月异的速度发展,因此计算机及相关专业的学生学习和掌握数据库知识是非常必要的。

当前,有关数据库教材均存在以下两个问题:一是内容更新速度慢,不能适应数据库技术的发展;二是教材未能适应面向地方的应用型本科人才培养要求。经过多年课程建设、工程实践以及教学改革的探索,我们逐步形成数据库课程“以能力的培养为中心、案例驱动教学、边讲边练、讲练结合”的教学模式,结合教育部本科教学水平评估的要求,组织编写了本教材。

本教材是面向计算机专业学生学习数据库知识而编写的,其特点是内容全面,既包括数据库的基础理论知识,又包括数据库的前端和后端的应用技术。同时本书特别加强解决实际问题的内容,包括在数据库管理系统中对索引的管理方法以及如何构建提高数据查询效率的索引,如何编写带参数的存储过程以及如何自定义函数以实现复杂的数据查询功能等。

本书共分10章:第1章主要简述数据库的发展和数据管理的相关概念;第2章重点讲述关系数据库的理论基础、关系代数等;第3章以直观的图形演示SQL Server 2008的安装和相关环节的操作过程;第4章重点讲述SQL语句的相关语法;第5章主要讲述关系数据库理论和规范化的标准;第6章重点讲述数据库安全的概念和相关安全措施;第7章讲解数据库设计的理论;第8章主要讲述T-SQL的程序设计和相关数据库对象的设计实现;第9章简述当前数据库技术的发展和特点,为学生进一步学习数据管理和数据应用奠定基础;第10章以图书管理系统为基础讲解了数据库系统设计的全过程,该章可以作为课程设计的样本,指导学生的实践教学。另外,本书还附有部分实验指导,实验内容可根据实际情况分别予以调整。

本书编写人员有何宗耀、吴孝丽、薛冰、苏靖枫、张向娟、王斌斌、李忠、杜小杰、张妍琰、王刚、田慧和冯巧娟等,何宗耀、吴孝丽任主编。

因成书时间仓促,加之作者水平所限,书中有不妥之处,恳请广大同仁给予批评指正。

编 者

2014年3月

目 录

第 1 章 数据库概论	1
1.1 数据库技术的产生和发展	1
1.2 数据库系统的基本概念	4
1.3 数据库系统的内部系统结构	6
1.4 数据模型	8
1.5 最常见的四种数据模型	12
本章小结	22
习题	22
第 2 章 关系数据库	23
2.1 关系模型	23
2.2 关系代数	26
2.3 关系演算	33
本章小结	34
习题	34
第 3 章 SQL Server 2008 的安装与配置	38
3.1 SQL Server 2008 简介	38
3.2 SQL Server 2008 的安装	39
3.3 SQL Server Management Studio 管理器	43
3.4 T-SQL 与查询编辑器	50
本章小结	52
习题	52
第 4 章 关系数据库标准语言 SQL	53
4.1 SQL 语言概述	53
4.2 创建与管理数据库	54
4.3 数据表及其操作	70
4.4 强制数据完整性	81
4.5 数据查询	88
4.6 视图	104
4.7 索引	113

本章小结	119
习题	120
第5章 关系数据库理论	122
5.1 规范化问题的提出	122
5.2 函数依赖	123
5.3 关系模式的范式	128
5.4 关系模式的分解	132
5.5 关系模式的规范化	136
本章小结	138
习题	138
第6章 数据库保护机制	141
6.1 事务机制	141
6.2 数据库的安全性	143
6.3 并发控制与封锁	155
6.4 数据库恢复	162
6.5 数据的备份和还原	165
本章小结	168
习题	169
第7章 数据库设计	170
7.1 数据库设计概述	170
7.2 需求分析	173
7.3 概念结构设计	177
7.4 逻辑结构设计	181
7.5 物理结构设计	184
7.6 数据库的实施和维护	186
本章小结	188
习题	188
第8章 Transact-SQL 程序设计	189
8.1 Transact-SQL 程序设计基础	189
8.2 常量、变量、函数	191
8.3 流程控制	197
8.4 存储过程	200
8.5 触发器	210
8.6 游标的使用	221
本章小结	225

习题.....	225
第 9 章 数据高级处理技术.....	227
9.1 数据库系统的新发展	227
9.2 数据仓库与联机分析处理技术	231
9.3 数据挖掘	234
9.4 分布式数据库	236
9.5 XML 技术	239
9.6 大数据	242
本章小结.....	244
习题.....	244
第 10 章 图书管理系统数据库设计实例	245
10.1 数据库设计原则和技巧.....	245
10.2 图书管理系统数据库设计实例.....	247
本章小结.....	266
附录.....	267
实验指导.....	278
参考文献.....	290

第1章 数据库概论

当今社会是一个信息化的社会,信息已经成为社会上各行各业的重要资源。数据是信息的载体,而数据库是相互关联的数据集合。数据库技术能够存储和管理大量复杂的数据,快速而有效地为多个不同的用户和应用程序提供数据,帮助人们有效利用数据资源。目前,数据库应用已遍及生活中的各个角落,例如学校的教学管理系统、图书馆的图书借阅系统、车站及航空公司的售票系统、电信局的计费系统、超市的售货系统、银行的业务系统、工厂的管理信息系统等。

数据库技术已经成为先进信息技术的重要组成部分,是现代计算机信息系统和计算机应用系统的基础和核心;因此,掌握数据库技术是全面认识计算机系统的重要环节,亦是适应信息化时代的重要基础。

本章主要介绍数据库系统的基本概念、数据模型、数据库系统结构、关系理论和数据库系统体系结构。

1.1 数据库技术的产生和发展

数据库技术是应数据处理和数据管理任务的需要而产生的。在应用需求的推动下,随着计算机硬件和软件的发展,数据管理技术经历了人工管理、文件系统和数据库系统三个阶段。

1.1.1 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。当时的硬件状况是:外存只有纸带、卡片、磁带,没有磁盘等直接存取的存储设备;软件的状况是: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。数据管理处于人工管理阶段。

这个阶段数据管理的特点是:

(1) 数据不保存

因为计算机主要用于科学计算,一般也不需要长期保存数据,只是在要完成某一个计算或课题时才输入数据,不仅原始数据不保存,计算结果也不保存。

(2) 应用程序管理数据

数据需要由应用程序自己管理,没有相应的软件系统负责数据的管理工作。应用程序中不仅要规定数据的逻辑结构,而且要设计物理结构,包括存储结构存取方法、输入方式等,因此程序员负担很重。

(3) 数据不共享

数据是面向应用的,一组数据只能对应一个程序。当多个应用程序涉及某些相同的数据时,由于必须各自定义,无法相互利用、相互参照,因此程序与程序之间有大量的冗余

数据。

(4) 数据不具有独立性

数据的逻辑结构或物理结构发生变化后,必须对应用程序做相应的修改,这进一步加重了程序员的负担。

在人工管理数据阶段,程序与数据之间的一一对应关系可用图 1-1 表示。

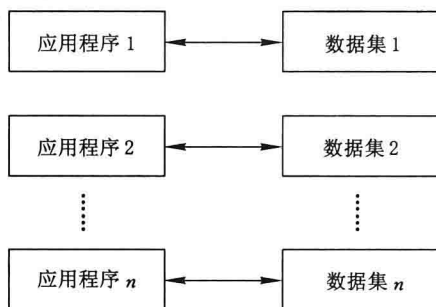


图 1-1 人工管理阶段程序与数据的对应关系

1.1.2 文件管理阶段

20 世纪 50 年代后期到 60 年代中期,这时硬件方面已有了磁盘、磁鼓等直接存取的存储设备;软件方面,操作系统中已经有了专门的数据管理软件,一般称为文件系统;处理方式上不仅有了批处理,而且能够联机实时处理。

文件系统仍然是一个不具有弹性的无结构的数据集合,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。

文件系统的最大特点是解决了应用程序和数据之间的一个公共接口问题,使得应用程序采用统一的存取方法来操作数据。在文件系统阶段数据管理的特点如下:

① 数据可以长期保留,数据的逻辑结构和物理结构有了区别,程序可以按文件名访问,不必关心数据的物理位置,由文件系统提供存取方法。

② 数据不属于某个特定的应用程序,即应用程序和数据之间不再是直接的对应关系,可以重复使用;但是文件系统只是简单地存取数据,相互之间并没有有机的联系,即数据存取依赖于应用程序的使用方法,不同的应用程序仍然很难有效地共享同一数据文件。

③ 文件组织形式多样化,有索引文件、链接文件和 HASH 文件等;但文件之间没有联系,相互独立,数据间的联系要通过程序去构造。

文件系统具有如下缺点:

① 数据冗余度大。文件与应用程序密切相关。相同的数据集合在不同的应用程序中使用时,经常需要重复定义、重复存储。例如:学生学籍管理系统中的学生情况、学生成绩管理系统的学生选课、教师教学管理的任课情况,所用到的数据很多都是重复的。这样相同的数据不能被共享,必然导致数据的冗余。

② 数据不一致性。由于相同数据重复存储、单独管理,给数据的修改和维护带来难度,容易造成数据的不一致。例如:学校在学生学籍管理系统中修改了某学生的情况,但在学生成绩管理系统中该学生相应的信息没有被修改,造成同一个学生的信息在不同的管理部门结果不一样。

③ 数据联系弱。文件系统中数据组织成记录,记录由字段组成,记录内部有了一定的结构;但是文件之间是孤立的,从整体上看没有反映现实世界事物之间的内在联系,因此很难对数据进行合理的组织以适应不同应用的需要。

1.1.3 数据库管理阶段

20 世纪 60 年代以来,计算机用于管理的规模越来越大,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。

这时硬件已有大容量磁盘,硬件价格下降;软件价格则上升,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,用户对联机实时处理的要求更多,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手段已经不能满足应用的需求,于是为解决多用户、多应用共享数据的需求,使数据为尽可能多的应用服务,数据库技术便应运而生,出现了统一管理数据的专门软件系统——数据库管理系统 DBMS。

从文件系统到数据库系统,标志着数据管理技术的飞跃。与人工管理阶段和文件管理系统阶段相比,数据库系统的特点主要有以下几个方面:

(1) 数据结构化

在文件系统中,尽管其记录内部已有了某些结构,但记录之间没有联系。而数据库系统则实现了整体数据的结构化,这是数据库的主要特征之一,也是数据库与文件系统的本质区别。

(2) 数据的共享性高、冗余度低、易于扩充

数据库系统从整体角度看待和描述数据,数据不再面向某个特定的应用程序,而是面向整个系统;因此,数据可以被多个用户、多个应用程序共享使用。数据共享可以大大减少数据冗余,节约存储空间。数据共享还能够避免数据之间的不相容性和不一致性。

(3) 数据独立性高

数据独立性包括数据的物理独立性和逻辑独立性。

物理独立性是指用户的应用程序与存储在磁盘上的数据库中的数据是相互独立的。也就是说,数据在磁盘上的数据库中如何存储是由 DBMS 管理的,用户程序不需要了解,应用程序要处理的只是数据的逻辑结构,这样一来当数据的物理存储结构改变时,用户的程序不用改变。

逻辑独立性是指用户的应用程序与数据库的逻辑结构是相互独立的,也就是说,数据的逻辑结构改变了,用户程序也可以不改变。

(4) 数据由 DBMS 统一管理和控制

数据库的共享是并发的(Concurrency)共享,即多个用户可以同时存取数据库中的数据,甚至可以同时存取数据库中的同一个数据。

DBMS 必须提供以下方面的数据控制功能:数据的安全性保护(Security);数据的完整性检查(Integrity);数据库的并发访问控制(Concurrency);数据库的故障恢复(Recovery)。

数据库管理阶段应用程序与数据之间的对应关系如图 1-2 所示。

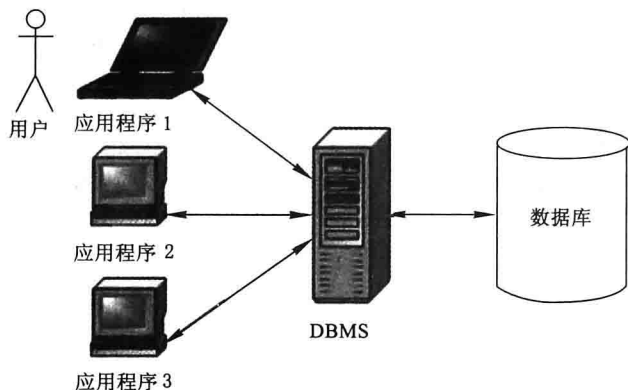


图 1-2 数据库管理阶段应用程序与数据之间的对应关系

1.2 数据库系统的基本概念

数据库是数据管理的新手段和新工具,使用数据库方法管理数据,可以保证数据的共享性、安全性和完整性。在学习数据库知识之前,首先介绍一些数据库最常用的术语和基本概念。

1.2.1 数据库的常见术语与基本概念

1.2.1.1 数据

数据(Data)是数据库中存储的基本对象。提起数据,大多数人头脑中的第一个反应就是能够进行加、减、乘、除运算以及各种统计计算的数值,如 14.56、56.99 等。其实数值只是最简单的一种数据,是数据的一种传统的和狭义的理解。从广义上讲,数据是对现实世界的抽象表示,是描述客观事物特征或性质的某种符号。描述事物的符号既可以是数字,也可以是文字、图形、图像、声音和语言等多种表现形式,它们都可以经过数字化后存入计算机。因此,从计算机数据管理的角度可更一般地将数据定义为:凡能够经过数字化处理进入计算机的符号都称为数据。

为了了解世界、交流信息,人们需要描述这些事物。在日常生活中直接用自然语言(如汉语)来描述。在计算机中为了存储和处理这些事物,人们通常将描述事物特征的若干数据项组成一个数据记录(Record)。数据项是数据的基本单元,即最小单位,它是对某类客观事物的某个特征或性质的数据抽象。每个数据项都有一个名字(称为数据项名)和可能的取值范围,称为数据项值域,简称为域(Domain)。

例如,在学生档案中,如果人们最感兴趣的是学生的学号、姓名、性别、年龄、所在系别、入学时间等数据,那么就先把它们组成为:

学生(学号,姓名,性别,年龄,所在院系,入学时间)

并将其称为记录型(Record Type),也称为记录的逻辑结构。它是对学生特征的一个抽象描述,当记录型中的每个数据项取确定的值时,例如:

(03051002,赵田,男,20,信息管理系,2010)

就成为一个记录。这个学生记录就是数据。

对于上面这条学生记录,了解其含义的人会得到如下信息:赵田是个大学生,男,20岁,2010年考入信息管理系;而不了解其语义的人则无法理解其含义。可见,数据的形式还不能完全表达其内容,需要经过解释。所以数据和关于数据的解释是不可分的,数据的解释是指对数据含义的说明,数据的含义称为数据的语义,数据与其语义是不可分的。

1.2.1.2 数据库

数据库(Data Base,简称DB)是长期储存在计算机内的、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组织、描述和储存,具有较小的冗余度、较高的数据独立性和易扩展性,并可为各种用户共享。

1.2.1.3 数据库管理系统

数据库管理系统(Database Management System,简称DBMS)是位于用户与操作系统之间的一层数据管理软件(图1-3),是数据库系统的核心组成部分;用户在数据库系统中的一切操作,包括数据定义、查询、更新及各种控制都是通过DBMS进行的。

DBMS有以下几个主要功能:

(1) 数据定义

DBMS提供数据定义语言DDL(Data Define Language),用户通过它可以方便地对数据库中的数据对象进行定义。例如,为保证数据库安全而定义的用户口令和存取权限,为保证正确语义而定义的完整性规则。

(2) 数据组织、存储和管理

DBMS要分类组织、存储和管理各种数据,包括数据字典、用户数据和数据的存取路径等;要确定以何种文件结构和存取方式在存储机上组织这些数据,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的利用率和方便存取,提供多种存取方法(索引查找、HASH查找和顺序查找等),以提高存取效率。

(3) 数据操纵

DBMS提供数据操纵语言DML(Data Manipulation Language),实现对数据库的基本操作,包括检索、插入、修改和删除等。SQL就是DML的一种。

(4) 数据库运行管理

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。DBMS通过对数据的安全性控制、数据的完整性控制、多用户环境下的并发控制以及数据库的恢复,来确保数据正确、有效,以及数据库系统的正常运行。

(5) 数据库的建立和维护功能

包括数据库的初始数据的装入、转换,数据库的转储、恢复、重组,系统性能监视、分析等功能。这些功能通常是由一些实用程序完成的。

(6) 其他功能

包括DBMS与网络中其他软件系统的通信功能;两个DBMS系统的数据转换功能;异构数据库之间的互访和互操作功能等。

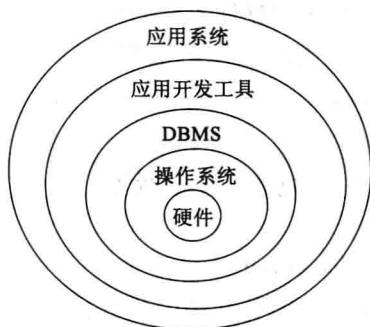


图1-3 DBMS在数据库系统中的地位

1.2.1.4 数据库系统

(1) 定义

数据库系统(Database System, 简称 DBS)是指在计算机系统中引入数据库后的系统,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。

(2) 数据库管理员

数据库管理员(Database Administrator, 简称 DBA),是一个负责管理和维护数据库服务器的人,负责全面管理和控制数据库系统。

(3) 数据库系统的构成

数据库系统由数据库、数据库管理系统 DBMS(及开发工具)、应用系统、数据库管理员(DBA)和用户构成,如图 1-4 所示。

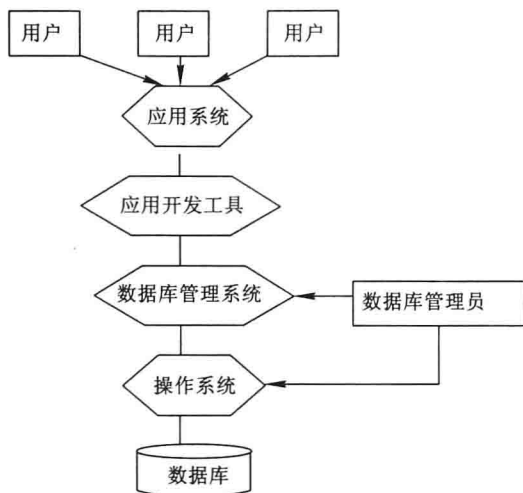


图 1-4 数据库系统的构成

1.3 数据库系统的内部系统结构

模式(Schema)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,而不涉及具体的值。模式的一个具体值称为模式的一个实例(Instance),同一个模式可以有多个实例。模式是相对稳定的,而实例则是相对变动的。

数据库管理系统在体系结构上通常都具有相同的特征,即采用三级模式结构,并提供二级映像功能。

1.3.1 数据库系统的三级模式结构

数据库系统在其内部具有三级模式和二级映像(图 1-5):三级模式分别是外模式、模式和内模式;二级映像则分别是外部级到概念级的映像以及概念级到内部级的映像,即外模式/模式映像和模式/内模式映像。

概念模式(Conceptual Schema)也称逻辑模式,是对数据库系统中全局数据逻辑结构的描述,是全体用户(应用)公共数据视图。它不涉及具体的硬件环境和平台,也与具体的软件环境无关。外模式(External Schema)也称子模式,它是数据库用户(包括应用程序员和最终用户)能够看见和使用的局部数据的逻辑结构和特征的描述,它是由概念模式推导而出来的,是数据库用户的数据视图,是与某一应用有关的数据的逻辑表示。一个概念模式可以有若干个外模式。内模式(Internal Schema)又称物理模式(Physical Schema),它给出了数据库物理存储结构和物理存取方法,如数据存储的文件结构、索引、集簇及 HASH 等存取方式与存取路径。内模式的物理性主要体现在操作系统及文件级上,它还未深入到设备级上(如磁盘及磁盘操作)。

模式的三个级别层次反映了模式的三个不同环境及它们的不同要求,其中:内模式处于最底层,它反映了数据在计算机物理结构中的实际存储形式;概念模式处于中间层,它反映了设计者的数据全局逻辑要求;而外模式处于最外层,它反映了用户对数据的要求。

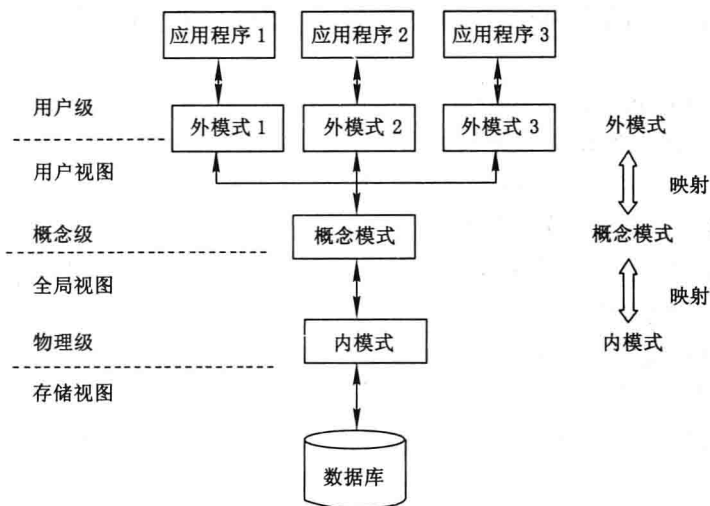


图 1-5 三级模式及二级映像关系图

1.3.2 数据库的二级映像与数据独立性

数据库系统的三级模式是对数据的三个级别的抽象视图的描述,使用户能逻辑地、抽象地处理数据,而不必关心数据在计算机中的具体表示方式和存储方式。

在数据库的三级模式结构中,数据库模式即全局逻辑结构是数据库的中心和关键,它独立于数据库的其他层次;因此,设计数据库模式时,应首先确定数据库的逻辑模式。数据库的内模式依赖于它的全局逻辑结构,但独立于数据库的外模式和具体的存储设备;它是将全局逻辑结构中所定义的数据结构及其联系按照一定的物理存储策略进行组织,以达到较好的时间和空间效率。数据库的外模式面向具体的应用程序,它定义在逻辑模式之上,但独立于内模式和存储设备;当应用需求发生较大变化,相应外模式不能满足其视图要求时,外模式就需要进行相应的修改,所以设计外模式时应充分考虑到应用的扩充性;不同的应用程序有时可以共用同一个外模式。

为了能够在内部实现这三个抽象层次的联系和转换,数据库管理系统在这三级模式之间提供了二级映像:外模式/模式映像和模式/内模式映像。正是这两层映像保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性。

(1) 外模式/模式映像

模式描述的是数据的全局逻辑结构,外模式描述的是数据的局部逻辑结构,对应于同一个模式可以有任意多个外模式。对于每一个外模式,数据库系统都有一个外模式/模式映像,它存在于外部级和概念级之间,用于定义用户的外模式与模式之间的对应关系。这些映像定义通常包含在各自外模式的描述中。

当模式改变(如增加新的关系、新的属性,改变属性的数据类型等)时,由数据库管理员对各个外模式/模式的映像做相应改变,可以使外模式保持不变。应用程序是依据数据的外模式编写的,因而不必修改,从而保证了数据与程序的逻辑独立性,简称数据的逻辑独立性。

(2) 模式/内模式映像

数据库中只有一个模式,也只有一个内模式,所以模式/内模式映像是唯一的,它定义了

数据库全局逻辑结构与存储结构之间的对应关系。该映像定义通常包含在概念模式的定义描述中。

当数据库的内模式存储结构改变了(如选用了另一种存储结构),由数据库管理员对模式/内模式映像做相应改变,可以使模式保持不变,从而使应用程序也不必改变,保证了数据与程序的物理独立性,简称数据的物理独立性。

三级结构和二级映像的优点如下:

① 保证数据的独立性。将模式和内模式分开,保证了数据的物理独立性;将外模式和模式分开,保证了数据的逻辑独立性。

② 简化了用户接口。按照外模式编写应用程序或敲入命令,而无须了解数据库内部的存储结构,方便用户使用系统。

③ 有利于数据共享。在不同的外模式下可有多个用户共享系统中数据,减少了数据冗余。

④ 有利于数据的安全保密。在外模式下根据要求进行操作,不能对限定的数据操作,保证了其他数据的安全。

1.4 数据模型

数据库中的数据以一定的数据模型进行组织、描述和存储,可以供用户共享,具有优化数据冗余度和独立性,完善数据存储、操作、保护和恢复等功能。

1.4.1 数据模型的概念

在数据库用数据模型这个工具来抽象、表示和处理现实世界中的数据和信息。通俗地讲,数据模型就是现实世界的模拟。

数据模型(Data Model)是专门用来抽象、表示和处理现实世界中的各种数据和信息的工具。计算机系统是不能直接处理现实世界的,现实世界只有数据化后,才能由计算机系统来处理这些代表现实世界的的数据。为了把现实世界的具体事物及事物之间的联系转换成计算机能够处理的数据,必须用某种数据模型来抽象和描述这些数据。现有的数据库系统均是基于某种数据模型的。因此,了解数据模型的基本概念是学习数据库的基础。

对不同使用对象和应用目的,应采用不同的数据模型。不同的数据模型使用模型化数据和信息的工具不同。根据模型的不同应用,可将模型分为两类,分别属于两个不同的层次:

第一类模型是概念模型,也称信息模型,介于现实世界与机器世界中间,是一种独立于计算机系统的数据模型,完全不涉及数据在计算机中的表示,只是用于描述某个特定组织所关心的数据结构。概念模型是按用户的观点对数据建模,强调其语义表达能力,概念应该简单、清晰、易于用户理解;它是对现实世界的第一层抽象,是用户和数据库设计人员之间进行交流的工具。这类模型中最常用的是实体联系模型。

第二类模型是数据模型,包括网状模型、层次模型和关系模型等,是以计算机系统的观点对数据建模,是直接面向数据库的逻辑结构,是对现实世界的第二层抽象。这类模型直接与 DBMS 有关,称为逻辑数据模型,简称逻辑模型,又称结构模型。其具有严格的形式化定义,便于定义和操纵数据,数据库语言包括无二义性的语法和语义。

1.4.2 实体联系模型

1.4.2.1 概念模型

人们对现实世界进行认识和抽象时,有时对数据结构形式只考虑其数据本身的结构和相互间的内在联系,而不考虑计算机的具体实现。

(1) 实体

实体(Entity)是现实世界中可以相互区别的事物或活动,如一个文件、一项活动等。

实体集(Entity Set)是同一类实体的集合。如一个班级的全部课程、一个图书馆的全部藏书、一年中的所有会议等都是相应的实体集。

实体型(Entity Type)是对同类实体的共有特征的抽象定义。如大学生的共有特征为姓名、性别、年龄、住址、专业、班级等。这几个特征定义了其实体型,每个学生都具有这些特征,但具体的特征值可以相同也可以不同,如姓名和年龄等。对于同一类实体,根据人们的不同认识和需要,可能抽取不同的特征,从而定义出不同的实体型。

实体值(Entity Value)是符合实体型定义的、对一个实体的具体描述。

实体、实体集、实体型、实体值等概念有时很难区分,在以后叙述中经常统称为实体,可根据上下文知其具体含义。

(2) 联系

联系(Relationship)是指实体之间的相互关系,它通常表示一种活动。如一张订单、一个讲座、一场比赛、一次选课等都是联系。在一张订单中涉及商品、客户(顾客)和销售员之间的关系,即某个客户从某个销售员手里订购某件商品。

联系集(Relationship Set)是同一类联系的集合。如一次展销会上的全部订单、一次会议安排中的全部讲座、一次比赛活动中的所有比赛场次、一个班级同学的所有选课等都是相应的联系集。

联系型(Relationship Type)是对同类联系的共有特征的抽象定义。

与实体的有关概念类似,联系、联系集、联系型、联系值等概念也常统称为联系。联系元数是指一个联系中所涉及的实体型的个数。若涉及两个实体型则称为二元联系,若涉及三个实体型则称为三元联系等。特殊地,若涉及的两个实体型对应同一个实体则为一元联系。

在选课联系中,涉及学生和课程两个实体,因此称为二元联系。一个联系涉及的实体由用户需求决定。如对于选课联系,最简单的为二元联系,它只涉及学生和课程;较复杂的为三元联系,它涉及学生、课程和教师三个实体。

实体与联系实际上无本质区别,都有相应的特征标识,都具有型和值的概念,只是含有较多的联系特征,如在选课联系中含有学号、课程号等联系特征,通过联系特征与其他实体发生联系。以后为了叙述方便,常把联系和实体统称为实体(或联系实体)。

(3) 属性、键和域

属性(Attribute)是描述实体或联系中的一种特征,一个实体或联系通常具有多个特征,需要多个相应属性来描述。实体选择的属性由实际应用需要决定,并非固有不变。如对于人事、财务部门都使用职工实体,但每个部门所涉及的属性不同:人事部门关心的是职工号、姓名、性别、出生日期、职务、职称、工龄等属性;财务部门关心的是职工号、姓名、基本工资、岗位津贴、内部津贴、交通补助等属性。

键(Key)或称码、关键字、关键码等,是实体间相互区别的一种唯一标识,如学号、身份

证号、工号、电话号码等。一个实体可以存在多个键。如在职工实体中,若包含职工编号、身份证号、姓名、性别、年龄等属性,则职工编号和身份证号都是键。键可能是实体中的一个或一组属性,特别是在联系实体中常为一组属性。如在学生或课程实体中,键分别是学号和课程号,所对应的都是单个属性;而在选课联系中,由学号和课程号联合表示的(学号,课程号)才能标识一个联系值。实体中用于键的属性称为主属性(Main Attribute),否则称为非主属性(Non-main Attribute)。如在职工实体中,职工号为主属性,其余为非主属性。

域(Domain)是实体中相应属性的取值范围。如性别属性域为(男,女)。

(4) 联系分类

联系分类(Relationship Classify)是讨论两个实体型(含联系型在内)之间的联系的类别。按照一个实体型中的实体个数与另一个实体型中的实体个数的对应联系,可分类为一对一联系、一对多联系、多对多联系三种情况。

① 一对一联系。若一个实体型中的一个实体至多与另一个实体型中的一个实体发生联系,同样另一个实体型中的一个实体至多与该实体型中的一个实体发生联系,则这两个实体型之间的联系被定义为一对一联系,简记为 1:1。如每个人只有一个身份证号。

② 一对多联系。若一个实体型中的一个实体与另一个实体型中的任意多个实体(含 0 个)发生联系,而另一个实体型中的一个实体至多与该实体型中的一个实体发生联系,则这两个实体型之间的联系被定义为一对多联系,简记为 1:n。如企业和客户之间就是一对多联系。与一对多联系相反的是多对一联系。一对多联系的两个实体也可以为同一个实体。如在一个职工表中,领导与被领导之间的联系。

③ 多对多联系。若一个实体型中的一个实体与另一个实体型中的任意多个实体(含 0 个)发生联系,反过来也一样,另一个实体型中的一个实体与该实体型中的多个实体(含 0 个)实体发生联系,则这两个实体型之间的联系被定义为多对多联系,简记为 m:n。

1.4.2.2 E-R 模型

E-R 模型(Entity-Relationship Model)是人们描述数据及其联系的概念模型,是数据库应用系统设计人员和普通非计算机专业用户进行数据建模和沟通与交流的有力工具,使用起来非常直观,简单易行。进行数据库应用系统设计时,首先要根据用户需求建立合乎需要的 E-R 模型,然后再建立与计算机数据库管理系统相适应的逻辑数据模型和物理数据模型,最后才能在计算机系统上安装、调试和运行数据库。

(1) E-R 模型中的基本构件

E-R 模型是一种用图形表示数据及其联系的方法,所使用的图形构件(元件)包括矩形、菱形、椭圆形和连接线。矩形表示实体,矩形框内写上实体名;菱形表示联系,菱形框内写上联系名;椭圆形表示属性,椭圆形框内写上属性名;连接线表示实体、联系与属性之间的所属关系或实体与联系之间的相连关系。

(2) 各种联系的 E-R 图

表示对于一对一、一对多和多对多三种联系对应的 E-R 图如图 1-6~图 1-8 所示。

下面用 E-R 图来表示教师实体、课程实体、学生实体之间的联系,如图 1-9 所示。学生实体与课程实体之间的联系为选修,教师实体与课程实体之间的联系为讲授。