



面向21世纪高等院校计算机系列规划教材
COMPUTER COURSES FOR UNDERGRADUATE EDUCATION

数据库系统原理及应用



杨毅 主编



科学出版社

www.sciencep.com

数据库系统原理及应用

杨毅 主编

何红玲 胡坤融 副主编

科学出版社

北京

内 容 简 介

本书以 SQL Server 2000 为背景,系统介绍了数据库系统的基本概念、原理和应用开发技术。内容包括:数据模型与数据库系统模式结构,关系模型与关系代数运算,关系数据库标准语言 SQL,关系模式的规范化理论,数据库设计,数据库安全与保护,数据库访问技术,数据库技术的新发展等。书中有较多例子和适量习题,数据库设计部分有较完整的实例。有关章节配有上机实验题,有利于教师教学和学生学习。

本书可作为高等院校计算机专业本、专科学生的数据库课程教材或参考书,也可供从事数据库教学与科研工作的教师和科技工作者参考。

图书在版编目(CIP)数据

数据库系统原理及应用/杨毅主编. —北京:科学出版社,2004
ISBN 7-03-014363-9

I.数… II.杨… III.数据库系统 IV.TP311.13

中国版本图书馆 CIP 数据核字(2004)第 093969 号

策划编辑:李振格 熊盛新/责任校对:柏连海

责任印制:吕春珉/封面设计:东方人华平面设计部

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2004年9月第一版 开本:787×1092 1/16

2004年9月第一次印刷 印张:18 1/2

印数:1—3 000 字数:425 000

定价:27.00 元

(如有印装质量问题,我社负责调换<双青>)

前 言

数据库技术自 20 世纪 60 年代中期产生以来,已历经三代。数据库管理系统 DBMS 成为各种应用系统开发的基础平台。在高等院校中,数据库原理及应用课程的重要性也越来越突出,它不仅是计算机、信息管理等专业的主干课程,而且还成为许多非计算机专业学生的必修或选修课程。

数据库技术有完善的理论基础,同时又是应用性极强的课程,所以,高校计算机或信息管理类专业的大学生,必须兼顾数据库理论与实际的应用。我们注意到,现有的很多数据库教材在内容和体系结构上有两种倾向:一种是偏重于理论,较少注意学生在学习时的实验操作等应用内容;另一种是较系统地介绍某一种 DBMS,突出信息系统软件的编码开发,由于篇幅有限,理论方面的介绍必然不够深入,使应用教学低层次化。

本书力图较好地将理论与实践结合起来,一方面,将数据库系统理论的核心内容进一步精细提炼并较完整保留,在叙述理论内容时配备较丰富的例题;另一方面,以常用的 SQL Server 2000 作为 DBMS,将涉及到的主要功能分散贯穿至各个章节,既克服了对 DBMS 系统介绍所带来的多余篇幅,又保证了学生实际操作与练习的需要。而且,后面有关章节还配有实验,教学实习可按照实验内容有序进行,具有更强的适应性,有利于教师的教学和学生的学习。另外,通过介绍完整具体的数据库设计实例,将数据库理论的各种知识有机地结合起来,为学生架起一座从理论到实践的桥梁。

全书共分 8 章。第 1 章为绪论,对数据管理技术和数据库技术的发展、常用数据模型、数据库的模式结构等内容进行了介绍;第 2 章为关系数据库,主要介绍了关系数据库模型的基本概念和关系代数查询语言。考虑到常用的 3 种关系查询语言(关系代数、元组关系演算、域关系演算)功能上的等价性,本章仅介绍关系代数运算。第 3 章为关系数据库标准语言 SQL,全面详细地介绍了 SQL 数据定义、数据查询、数据更新、数据控制、嵌入式 SQL 及查询优化等有关知识。第 4 章为关系数据库规范化理论,主要内容是函数依赖、多值依赖及关系模式的范式(1NF~4NF),画*号的章节作为选学内容。第 5 章为数据库的安全与保护,主要介绍了数据库安全、数据库恢复、并发控制、完整性控制等 4 个方面内容。第 6 章为数据库设计,介绍了数据库设计中需求分析、概念设计、逻辑设计、物理设计、数据库实施、数据库运行维护等 6 个方面的内容。第 7 章为数据库访问技术,介绍了普遍使用的 ODBC 和 ADO。第 8 章为数据库技术的新发展,主要介绍了面向对象数据库、分布式数据库和数据仓库等新一代的数据库技术。讲授本书内容大约需要 52~60 学时,实习为 12~16 学时,总学时为 64~76 学时。有“*”的内容对专科或学时较少的本科生不作要求。

本书写作时注意将内容多方位、多角度地讲解,例题丰富,层次分明,概念清楚,逻辑性强,理论叙述深入浅出,实际应用完整具体。可作为高等院校计算机专业本、专科学子的数据库课程教材或参考书,也可供从事数据库教学与科研工作的教师和科技工

作者参考。

本书主要由杨毅、何红玲、胡坤融、王凤军、张悦编写，杨毅担任主编并草拟提纲、规划各章节内容。其中第1、4章由杨毅负责编写，第2章由王凤军编写，第3、7章由何红玲编写，第5、8章由胡坤融编写，第6章由张悦编写。杨毅完成全书的修改和统稿工作，并重点对第6、7章进行了修改补充。

由于水平有限，疏漏和不当之处恳请广大师生和读者批评指正。

目 录

第 1 章 绪论.....	1
1.1 数据管理概述.....	1
1.1.1 信息、数据与数据管理.....	1
1.1.2 数据管理技术的发展.....	2
1.1.3 数据库技术的发展.....	6
1.2 数据模型.....	8
1.2.1 数据模型的层次结构.....	8
1.2.2 数据模型的组成三要素.....	9
1.2.3 概念模型.....	10
1.2.4 常用数据模型.....	15
1.3 数据库系统的模式结构.....	21
1.3.1 数据库系统模式的概念.....	21
1.3.2 数据库系统的三级模式结构.....	21
1.3.3 数据库的二级映像.....	23
1.4 数据库系统的组成.....	24
习题.....	26
第 2 章 关系数据库.....	28
2.1 关系模型概述.....	28
2.2 关系的数据结构和定义.....	29
2.2.1 关系.....	29
2.2.2 关系模式.....	33
2.2.3 关系数据库.....	34
2.3 关系的完整性.....	34
2.4 关系运算.....	35
2.4.1 传统的集合运算.....	36
2.4.2 专门的关系运算.....	38
2.4.3 关系运算举例.....	44
习题.....	45
第 3 章 关系数据库标准语言 SQL.....	47
3.1 SQL 概述.....	47
3.1.1 SQL 发展简述.....	47

3.1.2 SQL 的特点.....	47
3.2 SQL 数据定义.....	48
3.2.1 SQL 模式的创建和撤销.....	48
3.2.2 基本表操作.....	49
3.2.3 索引操作.....	51
3.2.4 SQL Server 中数据定义的实现.....	53
3.3 SQL 数据查询.....	54
3.3.1 单表查询.....	56
3.3.2 连接查询.....	61
3.3.3 嵌套查询.....	64
3.3.4 集合查询.....	70
3.3.5 其他形式的查询.....	71
3.3.6 SQL Server 中数据查询的实现.....	75
3.4 SQL 数据更新.....	78
3.4.1 插入数据.....	78
3.4.2 修改数据.....	79
3.4.3 删除数据.....	79
3.4.4 SQL Server 中数据更新的实现.....	80
3.5 SQL 视图.....	81
3.5.1 定义视图.....	81
3.5.2 更新视图.....	84
3.5.3 SQL Server 中的视图操作.....	85
3.6 SQL 数据控制.....	87
3.6.1 授权.....	88
3.6.2 回收权限.....	90
3.6.3 SQL Server 中数据控制的实现.....	90
3.7 嵌入式 SQL.....	91
3.7.1 嵌入式 SQL 介绍.....	91
3.7.2 嵌入式 SQL 语句与主语言之间的通信.....	92
3.7.3 在 Visual Basic 中的嵌入式 SQL 语句.....	94
3.8 查询优化.....	98
3.8.1 优化概述.....	98
3.8.2 查询实例分析.....	99
3.8.3 优化的一般策略和步骤.....	101
习题.....	104
实验一 简单的 SQL 查询 (1).....	106
实验二 简单的 SQL 查询 (2).....	108
实验三 学生表的维护.....	109

第 4 章 关系数据规范化理论	111
4.1 问题的提出	111
4.1.1 存在异常的关系模式	111
4.1.2 异常原因分析	112
4.1.3 异常问题的解决	113
4.2 函数依赖	114
4.2.1 函数依赖的一般概念	114
4.2.2 候选码与主码	115
4.3 关系模式的规范化	116
4.3.1 范式及其类型	116
4.3.2 第一范式 (1NF)	116
4.3.3 第二范式 (2NF)	117
4.3.4 第三范式 (3NF)	118
4.3.5 BC 范式 (BCNF)	119
*4.3.6 多值依赖	121
*4.3.7 第四范式 (4NF)	125
4.3.8 关系模式规范化步骤	126
*4.4 函数依赖推理	127
4.4.1 函数依赖的逻辑蕴涵	127
4.4.2 Armstrong 公理系统	128
4.4.3 闭包的计算	130
4.4.4 函数依赖集的等价和覆盖	131
*4.5 关系模式的分解	133
4.5.1 模式分解中存在的问题	133
4.5.2 无损连接	134
4.5.3 保持函数依赖的分解	138
4.5.4 分解成 3NF 的模式集	139
4.5.5 关系模式设计原则	140
习题	140
实验 关系模式的分解与连接	143
第 5 章 数据库的安全与保护	145
5.1 数据库安全与保护概述	145
5.1.1 数据库安全问题的产生	145
5.1.2 数据库安全性目标	147
5.2 数据库的安全性	148
5.2.1 用户认证	148
5.2.2 存取权限	149

5.2.3	视图隔离	151
5.2.4	跟踪与审计	152
5.2.5	数据加密	152
5.2.6	SQL Server 的安全性管理	153
5.3	数据库的完整性保护	157
5.3.1	完整性约束的分类	157
5.3.2	完整性控制	158
5.3.3	存储过程和触发器	162
5.3.4	SQL Server 的完整性策略	169
5.4	数据库的并发控制技术	172
5.4.1	事务	172
5.4.2	数据库的并发问题	174
5.4.3	并发控制方法	175
5.4.4	并发调度的可串行性	178
5.4.5	两段锁协议	180
5.4.6	SQL Server 的并发控制	180
5.5	数据库的恢复技术	182
5.5.1	数据库的故障种类	182
5.5.2	恢复技术	183
5.5.3	SQL Server 的数据恢复技术	186
	习题	187
	实验 数据库安全管理	188
第 6 章	数据库设计	194
6.1	数据库设计概述	194
6.1.1	数据库和信息系统	194
6.1.2	数据库设计的方法和特点	195
6.1.3	数据库设计的一般步骤	197
6.2	需求分析	199
6.2.1	需求分析的任务	199
6.2.2	需求分析的步骤	200
6.2.3	数据字典	202
6.3	概念结构设计	205
6.3.1	概念设计基本方法	205
6.3.2	主要设计步骤	207
6.4	逻辑结构设计	214
6.4.1	E-R 模式到关系模式的转换	214
6.4.2	关系模式的优化	218

6.5 物理设计及实施	219
6.5.1 关系数据库物理设计	220
6.5.2 关系数据库的实施	223
6.5.3 SQL Server 的物理设计	224
6.6 数据库设计实例	226
习题	232
实验 数据库设计与实现	234
第 7 章 数据库访问技术	236
7.1 ODBC	236
7.1.1 ODBC 结构	236
7.1.2 数据源	237
7.1.3 ODBC API 使用基础	243
7.2 ADO	246
7.2.1 OLE DB 概述	247
7.2.2 ADO 概述	248
习题	252
第 8 章 数据库技术的新发展	253
8.1 面向对象数据库技术	253
8.1.1 面向对象方法简介	253
8.1.2 面向对象数据模型	254
8.1.3 面向对象数据库管理模型	256
8.2 分布式数据库	269
8.2.1 分布式数据库系统的产生和定义	269
8.2.2 分布式数据库系统的模式结构	271
8.2.3 分布式数据库管理系统及分类	272
8.3 数据仓库技术	273
8.3.1 数据的事物处理与分析处理	273
8.3.2 数据仓库的定义及特征	274
8.3.3 数据仓库系统的组成	275
8.3.4 数据仓库中数据的组织	276
8.3.5 数据仓库工具与数据挖掘工具	279
习题	285
主要参考文献	287

第1章 绪 论

信息爆炸式增长是现代信息社会最为典型的特征。在这个社会中，一个组织的成功比以往更加依赖于它准确及时地获取数据的能力，有效管理数据的能力，以及使用数据进行分析并指导其行为的能力。信息资源在向管理提出挑战的同时，也成为各个部门最重要的财富和资源。如果没有管理大量数据和快速找到与问题相关信息的能力，那么，随着信息量的剧增，信息将成为负担。而如果能有效处理和利用这些数据，信息将成为财富，成为一个企业或部门科学管理和决策的基石。

数据库是数据信息管理的核心技术，是计算机科学的重要分支，也是发展最为迅速的分支之一。20世纪60年代末，数据库技术初露头脚。今天，数据库技术已在企业管理信息系统（MIS）、计算机集成制造（CIMS）、计算机辅助设计与制造（CAD/CAM）、办公信息系统（OIS）、地理信息系统（GIS）等各个方面得到广泛应用。对于一个国家来说，数据库建设的规模、数据信息量的大小和使用频度、数据的安全可靠性已成为衡量这个国家信息化程度的重要标志。因此，数据库课程是计算机科学与技术专业、信息管理专业的重要专业课程之一。

1.1 数据管理概述

1.1.1 信息、数据与数据管理

信息是一个被广泛使用，同时又是一个非常模糊的概念。多数学者认为，信息是对客观世界中各种事物的变化和特征的反映，是客观事物之间相互作用和联系的表征，是客观事物经过感知或认识后的再现。从计算机数据管理的角度看，信息表现为数据。

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、除运算的数字，如1、3、7等。其实，数字只是一种最简单的数据，是对数据的一种传统和狭义的理解。一般将数据定义为：描述事物的符号记录。其外延非常广泛，可以是数字，也可以是文字、图形、图像、声音、语言等多种形式，它们都可以经过数字化后存入计算机。

数据是载荷信息的物理符号，它与信息的关系包含有数据处理的成分。从用户的观点看，数据表现为信息，从计算机的观点看，信息表现为数据。数据能表示信息，但并不任何数据都是信息。信息是数据处理的结果，是人们能够理解的、有用的数据。

例如，在工资管理系统中，如果人们感兴趣的是职工号，姓名，月份，等级工资，浮动工资，补贴，津贴，交通费，医保、公积金，个人所得税等，在计算机中，为了存储和处理这些事物，就要抽出对这些事物感兴趣的特征组成一个记录来描述，即可将这些信息组成如下的形式：

工资（工号，姓名，月份，等级工资，浮动，补贴，津贴，交通费，医保，公积金，个人所得税），这类信息可称为记录型（Record Type）信息，也称为记录的逻辑结构，它是对工资情况的一种抽象描述。这里，“工资”称为记录名，工号、姓名、等级工资等称为字段（field）。记录型中每个字段取确定的值时，就形成一个如下所示的职工工资记录：

(42, 刘佳, 2003-12, 662, 70, 105, 248, 10, 24.96, 65, 6.97)

对上面这条工资记录，了解其含义的人会得到如下信息：编号为 42 号的职工刘佳，2003 年 12 月份等级工资是 662 元，浮动 70 元，补贴 105 元，津贴 248 元，交通费 10 元，需交医保金 24.96 元，住房公积金 65 元，个人所得税 6.97 元。对于不了解其语义的人则无法理解其中的含义。可见，数据的组织形式还不能完全表达其内容，需要经过数据语义的解释。数据的语义就是数据的含义，数据与其语义是不可分的。数据的语义在用户的角度，表现为某种确定的信息。

数据管理包括对数据的收集、保存、检索、分类、分析、加工和传输等的全过程。

1.1.2 数据管理技术的发展

自 20 世纪 40 年代计算机问世以来，随着计算机软、硬件技术的迅速发展，其应用领域越来越广泛，从最初单纯的科学计算逐渐渗透到数据管理领域。现在，用于数据管理的计算机数量和用于科学计算的计算机数量之比超过 8:2，而且呈继续上升的趋势。在应用需求和计算机软、硬件技术快速发展的基础上，计算机数据管理技术不断更替和完善，其发展过程主要经历了 3 个阶段：

- (1) 人工管理阶段（20 世纪 50 年代中期以前）。
- (2) 文件系统阶段（20 世纪 50 年代后期到 20 世纪 60 年代中期）。
- (3) 数据库系统阶段（20 世纪 60 年代后期至今）。

1. 人工管理阶段

在这一阶段，计算机主要用于进行科学计算，硬件的情况是：内存小，价格昂贵，外存只有磁带、卡片、纸带等低速小容量顺序存储设备，没有磁盘等高速随机存储设备；软件情况是：没有操作系统，没有数据管理软件，以批处理方式处理数据。人工数据管理方式具有如下特点：

(1) 数据不专门保存。当时计算机主要用于科学计算，数据仅涉及少量输入数据，一般不需要将数据长期专门保存。通常是针对某一计算程序，运行时向内存输入数据，计算完成后就将内存释放，不仅用户数据，对系统软件有时也是这样处理的。

(2) 程序管理数据。由于没有专门的软件系统对数据进行管理，如果涉及到大量的数据处理问题，必须由程序员在应用程序设计时不仅考虑数据的逻辑结构，而且要设计物理结构，包括存储结构，存取方式，输入方式等。因而数据处理效率低且对程序员有很高的要求。

(3) 数据不能共享。数据共享（Data Sharing）是指多个用户、多个应用程序能够同时或不相互影响地使用同一个数据集合。但在人工管理阶段，数据面向应用，一个程

序对应一组特殊数据。即使多个程序涉及相同的数据，也必须各自定义和使用，无法相互利用和参照，程序之间存在大量数据冗余。

(4) 数据没有独立性。数据独立性是指数据和应用程序之间的一种依赖关系。由于必须在应用程序中对数据的逻辑结构及物理结构进行定义设计，故数据完全依赖于应用程序。一旦数据发生变化，需对应用程序代码进行修改，不仅加重了程序员负担，而且维护代价昂贵。

在这一阶段，数据和应用程序之间的关系如图 1.1 所示。

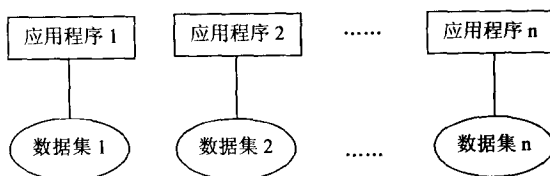


图 1.1 应用程序与数据之间的对应关系

2. 文件系统阶段

20 世纪 50 年代后期，随着应用需求的增加，计算机开始大量用于数据处理。这时，硬件已出现磁盘、磁鼓等速度较快的直接存取设备；软件有了操作系统，并在操作系统中配备了专门的数据管理软件——文件系统。数据处理方式不仅有批处理，而且能够联机实时处理。文件系统管理阶段具有以下特点：

(1) 数据可以长期保存。由于计算机大量用于数据处理，数据需要长期保存以便反复使用。文件系统即提供将数据以文件的形式长期保存在磁盘上的功能。

(2) 文件系统管理数据。操作系统提供的文件管理系统将数据组织成相互独立的数据文件，根据文件名进行访问，按照记录进行存取，文件的逻辑结构到物理结构的转换由文件系统自动完成，从而使程序和数据分开，数据与程序之间有了一定的独立性。这样，程序员可以将精力集中于数据的逻辑结构，不必或很少考虑其物理结构。数据在物理结构上的改变，通常不会反映在程序上，可大大节省程序维护的工作量。

文件系统管理数据比起手工管理有了很大进步，但仍存在如下缺点：

(1) 数据共享性差，冗余度大。数据冗余 (Data Redundancy) 是指同一组数据在多个文件中同时出现所引起的数据重复现象。在文件系统中，文件仍然是面向应用的，一个文件对应一个应用程序，当不同的应用程序具有部份相同的数据时，也必须建立各自的数据文件，而不能共享数据，所以数据重复存储，造成冗余浪费。同时由于相同数据独立管理，容易造成数据的不一致性，数据的修改和维护都很困难。

(2) 数据独立性差。由于数据文件面向某一特定应用，虽然对该应用来说，文件结构是优化的，但要增加一些新的应用将会很困难，系统扩充受到限制。另外，文件的逻辑结构在应用程序中定义，一旦数据的逻辑结构改变，必须修改应用程序，反之，应用程序改变，如使用不同的高级语言，也将引起文件数据结构和存储格式的改变。因此文件与程序之间独立性较差。

(3) 数据之间联系弱。不同的数据文件，各自管理自己的数据。数据之间缺乏相互关联的有效方法。即使文件之间的某些数据存在紧密的逻辑联系，但是由于实现的复杂性，很少在文件管理系统中提供这些数据之间联系的方法，因此，难以反映客观世界中事物之间复杂的内在联系。

在文件系统阶段，程序与数据文件之间的关系如图 1.2 所示。

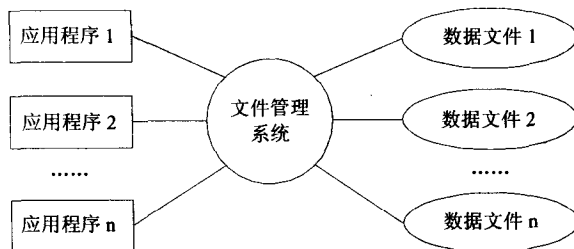


图 1.2 文件系统阶段应用程序与数据文件对应关系

3. 数据库系统阶段

20 世纪 60 年代后期，计算机技术发展迅速，硬件价格大大下降，出现了大容量磁盘，为保存和加工大量的信息提供了有利的硬件环境。越来越多的计算机从科学计算转向数据处理，对数据共享的要求日益强烈，对数据处理方式的要求也由批处理方式转向联机实时处理，并开始提出和考虑分布式处理。在这种情况下，以文件系统作为数据管理手段已经不能满足应用的需求，为了满足多用户、多应用共享数据的需求，数据库技术便应运而生，并出现了统一管理数据的专门软件系统——数据库管理系统（DataBase Management System，简称 DBMS）。

与文件系统相比，数据库系统对数据的管理实现了质的飞跃。数据库系统具有如下特点：

(1) 数据结构化。在文件系统中，虽然记录内部是有结构的，但在整体上是无结构的，即各个数据文件之间彼此是毫无联系的，要想实现应用程序对它们的交叉访问十分困难。而在数据库系统中，数据是按照某种数据模型组织，数据模型不仅描述数据本身的特征，而且还要描述数据之间的联系，从而实现整体结构化。数据库系统使数据不再面向特定的某个或某些应用，而是面向整个应用系统，且存取方式十分灵活，可以存取数据库中的某一个数据项、一组数据项、一个记录或一组记录。而在文件系统中，数据存取的最小单位是记录，访问粒度不能细到数据项。

例如，一个学校的选课管理系统需要记录学生信息、课程信息和选课信息，则需要 3 个记录型，如图 1.3 所示。

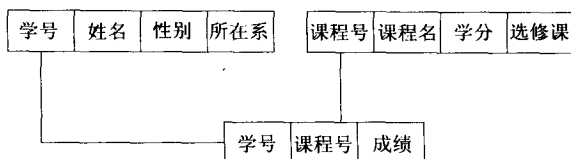


图 1.3 学生选课系统数据联系

如果用文件系统处理选课信息，需要建立 3 个数据文件。这 3 个文件彼此独立，没有联系。在应用中，要查询某人某课程的成绩，则同时涉及到 3 个文件，这时，程序员必须编写程序，将 3 个文件读到内存中，然后才能找到所需数据，处理过程十分复杂。如果用数据库技术，可以将学生信息、课程信息、选课信息建立 3 张数据表，表之间通过关键字学号、课程号联系起来，通过一个查询命令，即可得到某人某课程的成绩这样的信息。

(2) 数据的独立性较高。数据独立性指的是数据和应用程序之间的依赖关系。分为物理独立性和逻辑独立性。

物理独立性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，数据在磁盘上如何存取由数据库管理系统完成，用户程序不需要了解，更不需要特别处理。这样，当数据的物理存储结构改变了，应用程序也不用改变。

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，即数据的逻辑结构改变了，用户程序也可以不改变。

数据库系统中数据与程序的关系如图 1.4 所示，数据的独立性是由数据库管理系统的二级映像功能来保证的，这将在以后介绍。

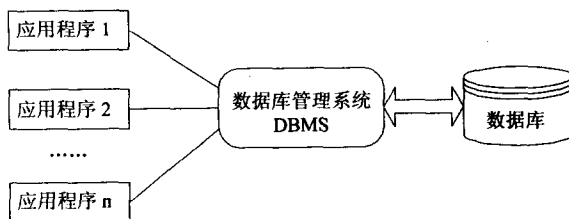


图 1.4 数据库系统阶段数据与程序的关系

数据与程序的独立性使得数据的定义从程序中分离出去，并且数据存取由数据库管理系统完成，从而简化了应用程序的编写，大大减少了应用程序的开发和维护费用，提高了程序开发和使用效率。

(3) 数据共享性高，冗余度低，易扩充。数据共享是指多种应用、多种语言相互覆盖地共享数据集合。在数据库阶段，数据不再分属于各个应用程序而是集中存放在数据库中面向整个系统。因此，数据可以被多个用户、多个应用共享。数据共享可以大大减少数据冗余，节约存储空间，还能避免数据之间的不相容性和不一致性。

所谓数据的不一致性是指同一数据的不同拷贝其值不相同。采用人工管理或文件管

理, 由于数据被重复存放, 当不同的应用使用和修改了不同位置的数据时, 往往造成数据的不一致甚至不相容。由于数据库集中存放, 统一使用, 故可最大限度地减少数据的不一致现象。

数据库中的数据是面向整个系统的有结构的数据, 不仅可被多个应用共享, 而且可以留下扩展应用接口, 这就使数据库系统具有易扩充的特性, 可以适应不断变化的各种用户群的要求。用户可以在各种数据子集上建立不同的应用系统, 当应用需求改变或增加时, 一般只要重新选取不同的子集或加上一部分数据便可形成新的应用。

(4) 对数据实行集中统一控制, 提供了完整的控制功能。数据库由于共享的特性使其在应用上存在多个用户同时使用的情况, 因而, 数据库管理系统须提供以下几方面的控制保护功能:

① 数据的安全性保护。数据的安全性是指保护数据以防止不合法的使用而造成数据的泄密和破坏。用户只能按规定, 对某些数据以某种方式进行处理。如用户身份检查和验证, 数据存取控制, 跟踪审计等。

② 数据的完整性控制。数据的完整性指数据的正确性、有效性和相容性。完整性控制是指将数据控制在有效范围内, 保证数据之间满足一定的关系。如日期只能是 1~31 之间的数字, 身份证号码是唯一的等。

③ 并发控制。当多个用户同时存取、修改数据库时, 可能会发生相互干扰而得到错误的结果或使得数据库的完整性遭到破坏, 因此, 数据库管理系统提供了对多用户的并发操作加以控制和协调的功能。如修改数据时必须对数据加锁, 修改完成后释放等。

④ 数据库恢复。计算机系统的硬件故障、软件故障、操作员的失误以及故意的破坏都会影响数据库中数据的正确性, 甚至造成部分或全部数据的丢失。数据库恢复即是数据库从错误状态恢复到某一已知的正确状态。

以上 4 个特征是数据库系统的主要特征。数据库系统的出现, 使得人们对信息处理方式发生改变。信息处理从以加工数据的程序为中心转变到以数据设计组织为中心的数据库系统方面。数据库系统对数据的管理, 有利于应用程序的开发和维护, 提高了数据的利用率, 开创了数据信息处理的新时代。

1.1.3 数据库技术的发展

数据库技术的萌芽出现在 20 世纪 60 年代初, 美国通用电气公司的 C.W.Bachman 是数据库领域的最早开拓者之一。1964 年, 他就领导设计了 IDS (Integrated Data Store) 系统, 从此开始了数据库技术研究和应用的不断发展。在数据库技术中, 始终贯穿一条发展主线, 即数据模型, 因此, 根据数据模型的发展变化将数据库技术的发展划分为 3 代。

1. 第一代数据库技术, 即层次型和网状型数据库技术时期

这一时期大约从 20 世纪 60 年代初到 20 世纪 70 年代初。数据库管理系统支持层次或网状数据模型, 其主要特点是支持三级模式结构; 用存取路径来表示数据之间的联系; 数据定义语言 (data definition language, DDL) 和数据操纵语言 (data manipulation

language, DML) 相对独立; 数据库语言采用过程性语言。这一时期较有影响的工作如下:

(1) 1964年, 美国通用电气公司的 C.W.Bachman 等人开发成功世界上第一个数据库管理系统 (DBMS) ——IDS 系统, 可为多个 COBOL 应用程序共享数据库, 奠定了网状数据库系统的基础。IDS 对早期数据库研究工作产生了重要影响, 为此, Bachman 于 1973 年获得了 ACM 图灵奖。

(2) 1969年, 美国 IBM 公司 Mcgee 等研制成功世界上第一个商品化 DBMS 系统——IMS (Information Management System) 系统, 这是第一个层次型的数据库管理系统产品。

(3) 1969~1970年, 美国 CODASYL (Conference On Data System Language) 研究会下属的 DBTG (DataBase Task Group) 对数据库方法进行了系统的研讨, 提出了 DBTG 报告, 建立了以网状数据模型为基础的数据库管理系统概念。

2. 第二代数据库技术, 即关系型数据库技术时期

这一时期主要从 20 世纪 70 年代到 20 世纪 90 年代。数据库管理系统支持关系数据模型。关系模型有着严格的数学理论基础, 概念简单, 易于用户理解和使用, 因而一经提出, 便迅速发展, 很快代替网状和层次数据库, 成为实用性最强的产品。系统主要特点是: 概念单一化, 数据及数据间的联系都用关系来表示; 以关系代数为理论基础, 数据独立性强; 数据库语言采用非过程化语言。这一时期具有代表性的工作有:

(1) 1970年, 美国 IBM 公司 San Jose 研究实验室的研究员 E.F.Codd 提出了关系数据模型, 开创了关系数据库系统的研究, E.F.Codd 因此在 1981 年获得了 ACM 图灵奖。

(2) 1974年, 美国 IBM 公司 San Jose 研究实验室研制成功 System R, 并在 IBM/370 大型机上运行, 这是世界上最早的关系数据库管理系统。以后又陆续推出了 SQL/DS、DB2 等商品化数据库系统产品。

(3) 1978年, 美国标准化组织发表关于数据库系统结构的最终报告——ANSI/X3/SPARC 建议, 规定了数据库系统的总体结构和特征。

(4) 1980年以后, 大、中、小型关系数据库管理系统 (relation DBMS, RDBMS) 产品迅速推出, 如 dBase、FoxBASE、Access、Oracle、Informix、Sybase 等。RDBMS 功能日益强大, 能够支持分布式数据库、客户/服务器数据库 (C/S) 以及客户/浏览器/服务器数据库 (B/S) 等, 同时实现了开放式网络环境下异种数据库的互联操作, 以及 OLTP (联机事物处理)、OLAP (联机事物分析) 的应用支持。

3. 第三代数据库技术, 即新一代数据库技术时期

它是面向对象模型为主要特征的数据库技术。这一代的数据库管理系统是基于扩展的关系数据模型或面向对象数据模型, 其主要特点是支持包括数据、对象和知识的管理; 在保持和继承第二代数据库技术基础上引入新技术, 如面向对向技术等, 对其他系统开放, 有良好的可移植性、可扩充性和可互操作性。

第三代数据库管理系统的代表性例子有: Servio 公司的 GemStone, OWTOS 公司的