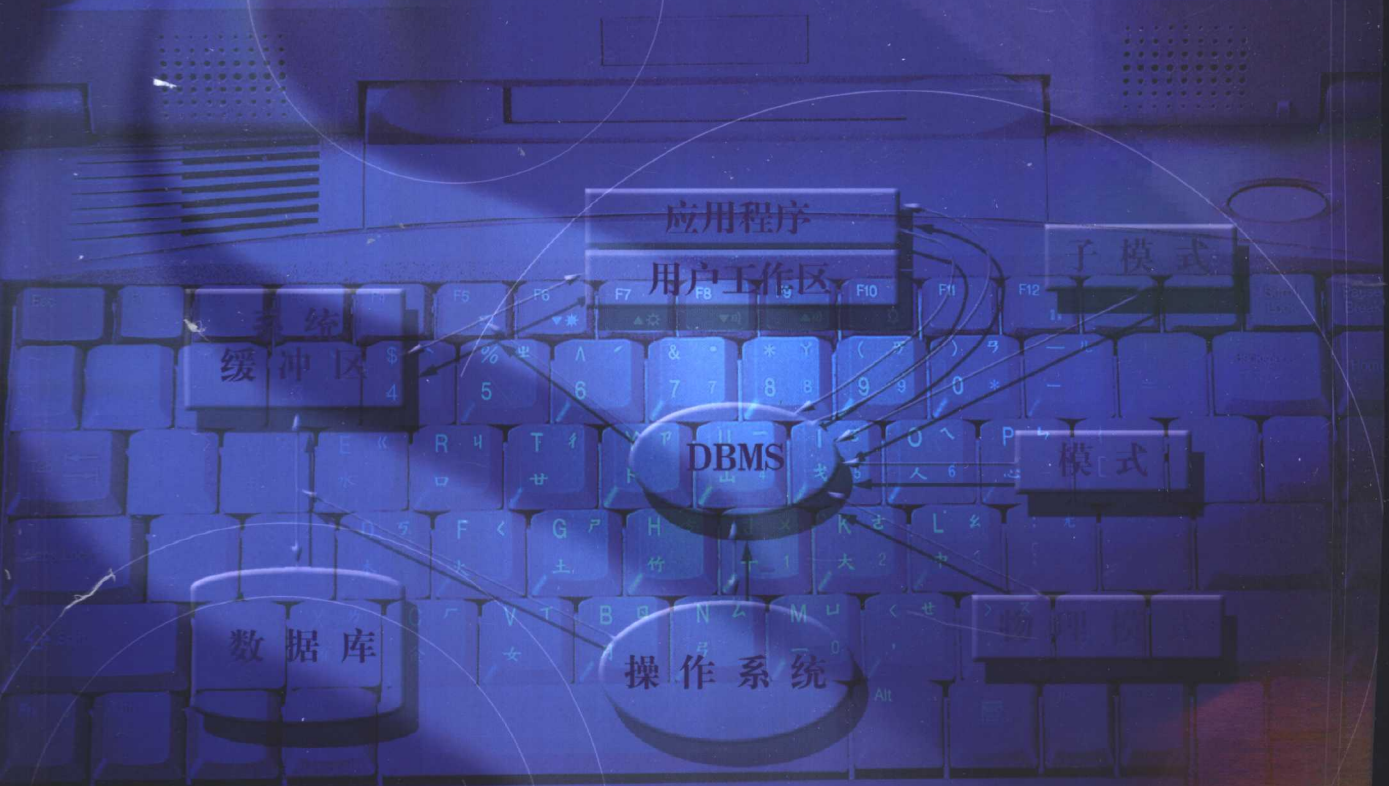




数据库 原理及应用

安迎建 等编



中国石化出版社

数据库原理及应用

安迎建 等编

中国石化出版社

内 容 提 要

本书系统讲述了当前数据库技术的基本原理及其应用实践,具体内容包括:数据库的基本概念、实体-联系模型、关系模型、SQL语言、关系数据理论与数据库设计、数据库的体系结构、事务管理、面向对象数据库和对象-关系数据库、数据库的存储及相关技巧等。在讲述基本理论和设计方法的同时,运用了大量的实例,对数据库的设计实践有很好的参考借鉴作用。

本书可作为大专院校计算机专业教学用书,也可供数据库设计人员阅读参考。

图书在版编目(CIP)数据

数据库原理及应用/安迎建等编.
—北京:中国石化出版社,2004
ISBN 7-80164-497-2

I. 数… II. 安… III. 数据库系统 IV. TP311.13

中国版本图书馆 CIP 数据核字(2004)第 004801 号

中国石化出版社出版发行

地址:北京市东城区安定门外大街 58 号

邮编:100011 电话:(010)84271850

读者服务部电话:(010)84289974

<http://www.sinopec-press.com>

E-mail:press@sinopec.com.cn

北京精美实华图文制作中心排版

河北省徐水县印刷厂印刷

新华书店北京发行所经销

*

787×1092 毫米 16 开本 14 印张 357 千字

2004 年 2 月第 1 版 2004 年 2 月第 1 次印刷

定价:28.00 元

前 言

数据库是目前数据管理的最常用技术,是计算机科学的重要分支。随着计算机技术的不断进步,数据库的普遍性及其重要性越来越被人们所重视。由于数据库具有数据结构化、最低冗余度、较高的程序与数据独立性、易于扩充、易于编制应用程序等特点,较大的信息系统都是建立在数据库设计之上的。对数据库的各种模型的应用也从一般管理扩大到计算机辅助设计、人工智能以至于科学计算等领域。

本书共分十章。第一章为数据库的基本概念,介绍数据库的一些基本知识,从第二章开始,以数据库设计过程为线索进行介绍。第二章为实体-联系模型,讲述了数据库设计的第一步:需求分析和概念数据库的设计。第三章讲述的是关系模型,讲述了数据库设计的第二步:逻辑数据库的设计。第四章是 SQL Server 概述,讲述了数据库设计的第三步:物理数据库的设计。第五章是 SQL 语言的内容,讲述了数据库中有关数据库、表、视图的操作语句。第六章为关系数据理论与数据库设计的内容,讲述了评判一个数据库是否科学合理的依据。第七章是事务管理,讲述了数据库管理和执行一些情况。第八章是数据库的体系结构,讲述了数据库系统的安全策略、整体管理理念等。第九章面向对象数据库和对象-关系数据库,第十章是数据库技术,讲述了数据库应用的一些技术、技巧。

读者通过对本书的学习能够了解并结合关系型数据库系统深入理解数据库系统的基本概念、原理和方法;掌握关系数据模型及关系数据语言,能熟练应用 SQL 语言表达各种数据操作;掌握 E-R 模型的概念和方法,关系数据库规范化理论和数据库设计方法;对数据库领域课题的深入研究有大致了解,激发在此领域中继续学习和研究的愿望,为学习数据库系统高级课程做准备。

本书根据数据库这门学科的特定规律编写。具有以下特点:

(一) 系统性。本教材以数据库设计的一般过程为主线,系统阐述了数据库的基本理论和设计方法。

(二) 技巧性。本书的举例和第十章,具体讲述了数据库的一些应用技巧,具有较高的参考价值。

(三) 衍生性。本书讲述了数据库的一些应用实例,对学生拓展视野,更好的

运用数据库奠定良好的基础。

总之,本书概念清晰、通俗易懂、实用性强,相信会得到广大读者的欢迎。

参加本书编写的有:黄国栋、蒋维平、刘帅、史海成、张玉川(以姓氏拼音排序)。

本书在出版过程中得到各位领导及同事的大力帮助,在此表示由衷的感谢。

本书如有不足之处,竭诚希望得到广大读者的批评指正。

编 者

目 录

第一章 数据库的基本概念	(1)
1.1 数据库发展的历史及现状	(1)
1.1.1 基本术语	(1)
1.1.2 数据管理的三个阶段	(2)
1.1.3 数据库技术发展的三个阶段	(5)
1.1.4 工程/科学领域数据处理的发展	(6)
1.1.5 数据库发展的新趋势	(7)
1.2 计算机集成化	(8)
1.2.1 在 CAX 中对 CIM 的要求	(8)
1.2.2 计算机集成化的方法	(9)
补充材料	(11)
练习题	(19)
第二章 实体 - 联系模型	(20)
2.1 基本概念	(20)
2.2 基本 E - R 图要点	(22)
2.2.1 基本要点	(22)
2.2.2 码在 E - R 图中的表示	(22)
2.2.3 角色表示要点	(22)
2.3 属性	(23)
2.3.1 属性的类型	(23)
2.3.2 属性在 E - R 图中的表示	(23)
2.4 映射约束	(24)
2.4.1 映射的基数 (Mapping Cardinalities)	(24)
2.5 弱实体集 (Weak Entity Set)	(26)
2.6 扩展 E - R 特性	(27)
2.7 E - R 图表示汇总	(29)
2.8 E - R 模型设计要点	(30)
2.9 概念数据库设计过程	(31)
2.10 E - R 模型向关系模式的转换	(32)
练习题	(35)
第三章 关系模型	(36)
3.1 数据视图和数据模式	(36)
3.1.1 数据库系统的三级模式结构	(36)
3.1.2 应用访问数据库数据的过程	(37)

3.2 数据库设计实例	(37)
3.2.1 实例模型	(37)
3.2.2 层次数据模型	(38)
3.2.3 网状数据库模型	(41)
3.2.4 关系模型	(44)
3.3 关系代数	(47)
3.3.1 选择运算	(48)
3.3.2 投影	(48)
3.3.3 并运算	(48)
3.3.4 差运算	(49)
3.3.5 交运算	(50)
3.3.6 广义笛卡尔积运算	(50)
3.3.7 更名运算	(51)
3.3.8 θ 连接	(51)
3.3.9 自然连接	(52)
3.3.10 除运算	(52)
3.3.11 除定义	(53)
3.3.12 赋值运算	(54)
3.3.13 广义投影	(55)
3.3.14 外连接	(55)
3.3.15 聚集函数	(56)
3.3.16 数据库修改(删除、插入、修改)	(56)
3.3.17 元组关系演算	(57)
3.3.18 域关系演算	(59)
练习题	(60)
第四章 SQL Server 数据库管理	(61)
4.1 SQL Server 概述	(61)
4.1.1 SQL Server 的特点	(61)
4.1.2 SQL Server 的主要服务	(62)
4.1.3 SQL Server 的客户软件	(62)
4.1.4 T-SQL(Transact-SQL)	(65)
4.2 SQL Server 的体系结构	(65)
4.2.1 SQL Server 的程序接口	(65)
4.2.2 数据对象接口	(65)
4.2.3 分层通信体系结构	(66)
4.3 SQL Server 安全策略	(66)
4.3.1 SQL Server 使用两级安全策略验证用户	(66)
4.3.2 SQL Server 服务器的安全模式	(66)
4.3.3 SQL Server 的特殊用户	(67)

4.4 SQL Server 数据库	(67)
4.4.1 SQL Server 数据库类型	(67)
4.4.2 系统数据库	(68)
4.4.3 用户数据库	(69)
4.4.4 SQL Server 数据库对象	(69)
4.4.5 创建用户数据库	(69)
4.4.6 创建数据库的方法	(70)
4.4.7 创建数据库的语句格式	(73)
4.4.8 创建数据库时的选项	(73)
4.4.9 创建数据库示例	(73)
4.4.10 删除数据库	(74)
4.5 创建和查询表	(75)
4.5.1 使用 T-SQL 命令创建表	(75)
4.5.2 Enterprise manager 创建表	(75)
4.5.3 在 Query Analyzer 查询表	(75)
练习题	(76)
第五章 数据库语言 SQL	(77)
5.1 SQL 特点	(77)
5.2 SQL 数据定义功能	(77)
5.2.1 域定义	(77)
5.2.2 基本表的定义	(78)
5.2.3 索引的定义	(80)
5.2.4 数据库的建立与撤销	(80)
5.2.5 SQL 数据定义特点	(80)
5.3 SQL 数据查询功能	(81)
5.3.1 SQL 数据查询基本结构	(81)
5.3.2 select 子句	(81)
5.3.3 重复元组的处理	(81)
5.3.4 from 子句	(82)
5.3.5 where 子句	(82)
5.3.6 更名运算	(82)
5.3.7 字符串操作	(83)
5.3.8 元组显示顺序	(83)
5.3.9 集合操作	(84)
5.3.10 分组和聚集函数	(84)
5.3.11 空值	(85)
5.3.12 嵌套子查询	(86)
5.3.13 派生关系	(89)
5.4 SQL 的数据修改功能	(90)

5.4.1 插入操作	(90)
5.4.2 删除操作	(90)
5.4.3 更新操作	(91)
5.5 视图	(92)
5.6 SQL 数据控制功能	(93)
5.6.1 安全性控制	(93)
5.6.2 完整性控制	(95)
5.6.3 SQL 中完整性约束	(98)
5.7 嵌入式 SQL	(99)
5.7.1 为什么使用嵌入式 SQL	(99)
5.7.2 嵌入式 SQL 执行过程	(99)
5.7.3 需要解决的几个问题	(100)
练习题	(102)
第六章 关系数据库理论及数据库设计	(104)
6.1 关系模式的设计问题	(104)
6.1.1 函数依赖	(105)
6.1.2 第一范式——1NF	(106)
6.1.3 第二范式——2NF	(107)
6.1.4 第三范式——3NF	(108)
6.1.5 BCNF	(108)
6.1.6 多值依赖 Vs 函数依赖	(110)
6.1.7 第四范式——4NF	(111)
6.2 函数依赖的推理规则	(112)
6.2.1 逻辑蕴涵	(112)
6.2.2 Armstrong 公理系统	(112)
6.3 模式分解	(117)
6.3.1 无损连接分解	(118)
6.3.2 保持函数依赖的分解	(121)
6.3.3 连接依赖	(127)
6.3.4 投影 - 连接范式(PJNF)	(128)
第七章 事务	(130)
7.1 事务概念	(130)
7.1.1 事务定义	(130)
7.1.2 事务特性(ACID)	(130)
7.2 事务状态	(131)
7.3 事务调度	(131)
7.3.1 事务的调度	(131)
7.3.2 并行 Vs 串行	(131)
7.3.3 事务执行示例	(132)

7.3.4 SQL 中一致性级别的定义	(133)
7.3.5 丢失修改	(133)
7.3.6 读脏数据	(134)
7.3.7 不能重复读	(134)
7.3.8 发生幻象	(134)
7.4 可恢复性	(134)
7.4.1 可恢复调度	(134)
7.4.2 无级联调度	(135)
7.5 可串行化	(135)
7.5.1 指令的顺序	(135)
7.5.2 冲突指令	(135)
7.5.3 冲突等价	(136)
7.5.4 冲突可串行化	(136)
7.5.5 视图等价	(136)
7.5.6 视图可串行化	(137)
7.5.7 冲突可串行化判定	(138)
7.5.8 视图可串行化判定	(139)
7.5.9 带标记的优先图的构造	(139)
练习题	(140)
第八章 数据库系统体系结构	(141)
8.1 集中式系统	(141)
8.2 客户-服务器系统	(142)
8.3 并行系统	(143)
8.3.1 并行数据库系统的特点	(143)
8.3.2 互连网络结构	(144)
8.3.3 并行数据库体系结构	(145)
8.4 分布式数据库系统	(146)
8.4.1 基本定义	(146)
8.4.2 D-DBS 的特点	(147)
8.4.3 D-DBS 的体系结构	(148)
8.4.4 分布式数据存储	(148)
8.4.5 分布式查询处理	(151)
8.4.6 通讯代价模型	(154)
8.5 多媒体数据库系统	(155)
8.5.1 多媒体数据库理论及技术支持	(155)
8.5.2 多媒体 DBMS 和媒体独立性	(155)
8.5.3 媒体 DBMS 应能支持	(155)
8.5.4 多媒体 DBMS 结构	(155)
8.5.5 多媒体数据库的存储结构	(155)

PT-36-1/5

第九章 面向对象数据库及对象 - 关系数据库	(157)
9.1 新的数据库应用	(157)
9.2 RDBMS 的局限	(157)
9.2.1 表达能力有限	(157)
9.2.2 类型有限	(158)
9.2.3 类型系统的任务	(158)
9.2.4 结构与行为的分离	(158)
9.2.5 版本与长事务	(158)
9.3 DBMS 分类矩阵	(159)
9.3.1 思考	(159)
9.3.2 矩阵分析	(159)
9.3.3 DBMS 需求	(160)
9.4 面向对象基本概念	(162)
9.4.1 对象	(162)
9.4.2 对象类	(162)
9.4.3 继承	(163)
9.4.4 多重继承	(164)
9.4.5 对象标识符(OID)	(165)
9.4.6 对象包含	(166)
9.4.7 面向对象的语言	(167)
9.5 持久化程序设计语言	(167)
9.5.1 关于持久化程序设计语言的概述	(167)
9.5.2 持久化 C++ 系统	(168)
9.6 对象关系数据库	(171)
9.6.1 概念数据模型(ODL)	(171)
9.6.2 对象 - 关系数据模型	(171)
9.6.2.1 嵌套关系	(171)
9.6.2.2 复杂类型与面向对象	(172)
9.6.2.3 有结构的类型和集合体类型	(172)
9.6.2.4 类型继承	(173)
9.6.2.5 表级继承	(174)
9.6.2.6 引用类型	(175)
9.6.2.7 与复杂类型有关的查询	(175)
9.6.2.8 解除嵌套	(176)
9.6.2.9 嵌套	(177)
9.6.2.10 函数	(177)
9.6.2.11 复杂值和复杂对象的创建	(177)
9.6.2.12 OO 数据库与 OR 数据库比较	(178)
9.6.2.13 E-R 模型	(178)

9.6.2.14 对象 - 联系图	(178)
9.6.2.15 BORDB 的定义语言	(179)
9.6.2.16 ORDB 的查询语言	(179)
第十章 数据库操作技术	(181)
10.1 并发控制	(181)
10.1.1 并发控制的例子	(181)
10.1.2 进程同步控制的基本概念	(184)
10.1.3 SQL Server 的加锁类型	(185)
10.2 数据库恢复	(185)
10.3 SQL Server 的事务处理过程	(187)
10.4 事务恢复和检查点	(187)
10.5 SQL Server 的数据备份与恢复	(188)
10.6 恢复系统和用户数据库	(190)
10.7 ADO 应用	(191)
10.8 产品数据管理(PDM)	(193)
10.9 几种不同的数据库连接示例	(195)
10.9.1 利用 ODBC 连接数据库	(195)
10.9.2 利用 ADO 数据控件连接数据库	(199)
10.9.3 利用 ADO 数据对象连接数据库	(201)
10.9.4 在 ASP 页面上利用 ADO 连接数据库	(202)
第十一章 数据库存储	(205)
11.1 物理存储介质	(205)
11.1.1 特性	(205)
11.1.2 介质分类	(205)
11.1.3 存储层次	(206)
11.1.4 磁盘性能估计	(206)
11.2 RAID	(207)
11.2.1 廉价磁盘冗余阵列(RAID)	(207)
11.2.2 RAID 级别	(208)

第一章 数据库的基本概念

本章概括介绍数据管理及数据库的发展过程,以及现代工程、科学领域对数据管理的要求和实现计算机集成的简单事例。

1.1 数据库发展的历史及现状

数据库系统是计算机系统的重要组成部分,是计算机软件发展的三大方向(数据库,网络和人工智能)之一。数据库是借助于计算机保存和管理大量复杂的数据和信息的软件工具,也可以理解成一个电子文件柜。数据库技术研究的主要问题是科学地组织和存储数据,如何高效地获取数据、更新数据和加工处理数据,并保证数据的安全性、可靠性和持久性。为了更好地了解和学习数据库,下面先介绍几个基本术语:

1.1.1 基本术语

数据(Data):反映客观世界的事实,并可以区分其特征的符号称之为数据。例如:字符、数字、文本、声音、图形、图像、图表、图片等,它们是现实世界中客观存在的,可以输入到计算机中进行存储和管理。

信息(Information):由原始数据经加工提炼而成的、用于决定行为、计划或具有一定语义的数据称为信息。

数据和信息之间的关系如同原料和成品,又具有相对性。同时还可以这样理解:“数据”表示在数据库中实际存在的内容,而用“信息”表示一些用户对这些数据的理解。

数据库 DB(Database):是现实世界中相互关联的大量数据及数据间关系的集合。例如家中存放衣物的衣柜,里边存放了大量的衣服,这就是数据,上衣的上面放的是裤子,裤子的上面放的是帽子,这就是数据之间的关系。因此可以得出数据库具有二要素:数据和数据之间的关系。

数据库管理系统 DBMS(Database Management System):是对数据库中的数据进行存储和管理的软件系统。包括存储、管理、检索和控制数据库中数据的各种语言和工具,是一套系统软件。也就是说,数据库管理系统在未安装数据库软件时就已经存在,是计算机系统本身就有的。

数据库系统 DBS(Database System):是对数据库和数据库管理系统的总称。是指相互关联的数据集合与操纵数据的软件工具集合。例如:SQL SERVER、ORACLE、ACCESS、FOXBASE、DB2、PARADOX等均称为数据库系统,属于应用软件。因此我们可以得出:DBS = DB + DBMS。有关以上几个概念的关系,请看图 1-1。

数据库应用程序接口 API(Application Programming Interface):是由 DBMS 为开发应用程序提供的操纵和访问数据库中数据的接口函数、过程或语言。例如:ODBC、BDE、ADO 等。

数据库应用程序 AP(Application Program):满足某类用户要求的操纵和访问数据库的程

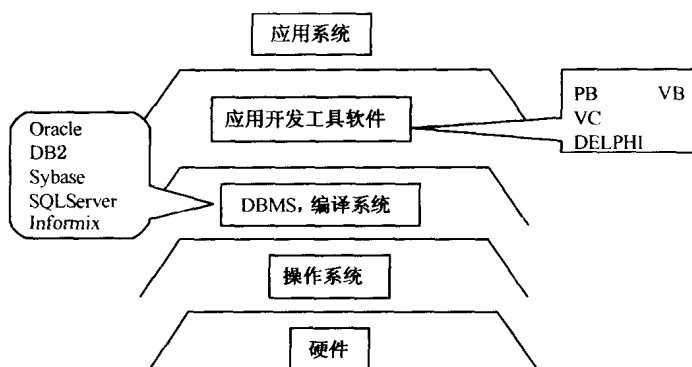


图 1-1 数据库系统的软硬件层次

序。例如：人事管理系统、ERP 等。

数据仓库(Data Warehouse)：一个数据仓库就是一个自带数据库的商业应用系统。(本章结束时详细阐述)

数据发掘(Data Mining)：数据发掘可帮助商业用户处理大量存在的数据，以期发现一些“意外的关系”，以便增加市场份额和利润。

信息发掘(Information Mining)：信息以多语言、多数据格式和大容量的形式出现。传统的信息组织方法(作者、主题和题目)不能发挥计算机按内容查找文档、连接文档和将同类文档归类的能力。信息发掘是指在文档、地图、照片、声音和影像之汪洋大海中发现相关信息，即面向内容的检索。这是个令人兴奋又富有挑战性的问题。

1.1.2 数据管理的三个阶段

所谓数据管理，是指如何用计算机管理数据。自从人类发明使用计算机处理数据以来，就有如何管理数据问题。在 40 年的计算机发展过程中，数据管理可划分为三个阶段：手工管理、文件管理和数据库系统管理。

1. 手工管理阶段

背景：

计算机主要用于科学计算：数据量小、结构简单，如高阶方程、曲线拟和等；外存为顺序存取设备：磁带、卡片、纸带，没有磁盘等直接存取设备；没有操作系统，没有数据管理软件：用户用机器指令编码，通过纸带机输入程序和数据，程序运行完毕后，由用户取走纸带和运算结果，再让下一用户上机操作，如图 1-2 所示。

特点：

- (1) 用户完全负责数据管理工作：数据的组织、存储结构、存取方法、输入输出等。
- (2) 数据完全面向特定的应用程序：每个用户使用自己的数据，数据不保存，用完就撤走。
- (3) 数据与程序没有独立性：程序中存取数据的子程序随着存储结构的改变而改变。
- (4) 磁带的优点：廉价地存放大容量数据。

缺点：顺序访问；1%所需，100%访问。

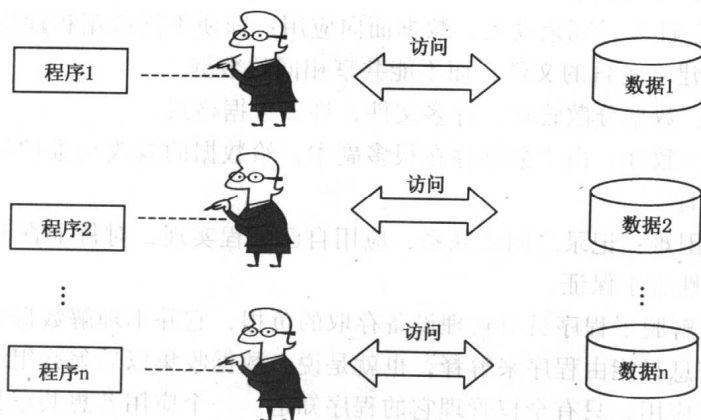


图 1-2 手工管理阶段示意图

2. 文件管理阶段

背景:

计算机不但用于科学计算,还用于管理。外存有了磁盘、磁鼓等直接存取设备。直接存取设备(DASD)(无须顺序存取,由地址直接访问所需记录)有了专门管理数据的软件,一般称为文件系统(文件存储空间的管理,目录管理,文件读写管理,文件保护,向用户提供操作接口),如图 1-3 所示。

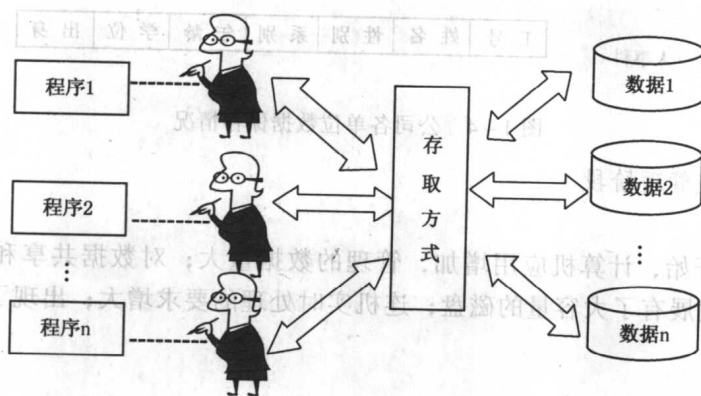


图 1-3 文件管理阶段示意图

特点:

(1) 系统提供一定的数据管理功能:存取方法(索引文件、链接文件、直接存取文件、倒排文件等);支持对文件的基本操作(增、删、改、查等),用户程序不必考虑物理细节;数据的存取基本上以记录为单位。

(2) 数据仍是面向应用的:一个数据文件对应一个或几个用户程序。

(3) 数据与程序有一定的独立性:文件的逻辑结构与存储结构由系统进行转换,数据在存储上的改变不一定反映在程序上。

(4) 数据与程序的独立性差:文件系统的出现并没有从根本上改变数据与程序紧密结合

的状况，数据的逻辑结构改变则必须修改应用程序。

(5) 数据的共享性差，冗余度大。数据面向应用：即使不同应用程序所需要的数据有部分相同时，也必须建立各自的文件，而不能共享相同的数据。

(6) 数据孤立：数据分散管理，许多文件，许多数据格式。

(7) 数据的不一致性：由于数据存在很多副本，给数据的修改与维护带来了困难，容易造成数据的不一致性。

(8) 数据查询困难：记录之间无联系，应用自己编程实现，对每个查询都重新编码。

(9) 数据完整性难于保证。

文件系统只是解脱了程序员对物理设备存取的负担，它并不理解数据的语义，只负责存储。数据的语义信息只能由程序来解释，也就是说，数据收集以后怎么组织，以及数据取出来之后按什么含义应用，只有全权管理它的程序知道。一个应用若想共享另一个应用生成的数据，必须同另一个应用沟通，了解数据的语义与组织方式。

以一个公司为例，每个单位都存有员工的信息，如工号、姓名、性别等这样就存在了数据冗余的现象。如图 1-4 所示。

劳资科

工号	姓名	性别	补贴
----	----	----	----

房产科

工号	姓名	性别	系别	住址
----	----	----	----	----

人事科

工号	姓名	性别	系别	年龄	学位	出身
----	----	----	----	----	----	----

图 1-4 公司各单位数据保存情况

3. 数据库系统管理阶段

背景：

60 年代后期开始，计算机应用增加，管理的数据量大；对数据共享和减少数据冗余要求强烈；硬件的发展有了大容量的磁盘；联机实时处理的要求增大；出现了解决数据共享的软件 DBMS。

特点：

(1) 具有面向多种应用的数据组织和结构

例如，学生数据库包括人事、学籍和健康。

(2) 具有高度的数据独立性

数据的存储结构和数据的逻辑结构之间是相互独立的，应用程序直接与数据的逻辑结构相关。

(3) 实现数据的高度共享并保证数据的完整性和安全性

(4) 高度的数据独立性

① 数据独立性：当数据的结构发生变化时，通过系统提供的映象(转换)功能，使应用程序不必改变。

② 数据的物理独立性：当数据的存储结构改变时，应用程序不必改变。

③ 数据的逻辑独立性：当数据的逻辑结构改变时，应用程序不必改变。

物理数据的独立性：全局逻辑数据结构独立于物理数据结构。

逻辑数据的独立性：指数据的全局逻辑结构独立于局部逻辑结构。

如图 1-5 所示，应用程序与数据的逻辑结构和物理存储结构之间的映射关系由 DBMS 完成。

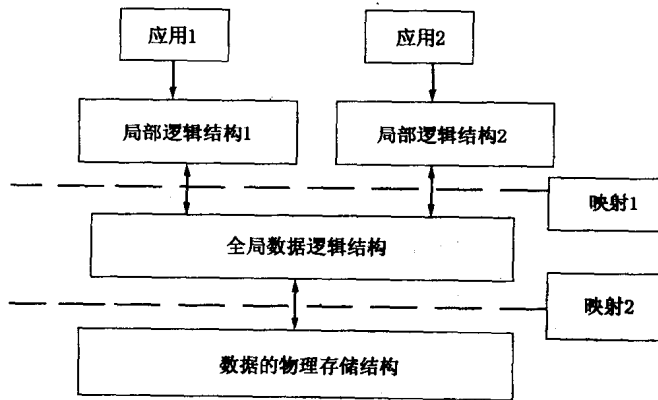


图 1-5 应用程序和数据库的映射

1.1.3 数据库技术发展的三个阶段

依数据模型划分数据库系统的分代：

第一代：层次和网状数据库(Hierarchical/Network)

IMS 层次数据库：层次数据库系统的代表产品是 IBM 在 1969 年研制出的 IMS(Information Management System)。层次数据库按层次组织数据，只能处理 1:1 和 1:N 的关系。如图 1-6 所示。

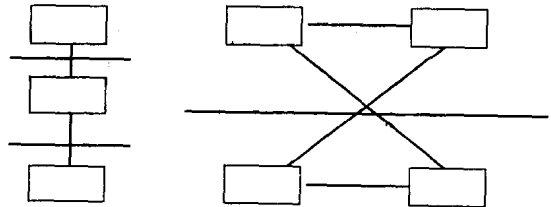


图 1-6 层次、网状数据库模型

网状数据库系统是以美国的 CODASYL (Conference On Data System Language) 的 DBTG

(Data Base Task Group) 报告为基础。报告规定了所有的术语和规范，系统提供：

数据定义语言 DDL (Data Description Language)

数据操纵语言 DML (Data manipulation Language)

可以处理 1:1, 1:N, M:N 的关系。

第二代：关系数据库系统 RDBMS(非过程化语言，很好的形式化基础)

关系数据库系统始于 20 世纪 70 年代末，由 E.F.Codd 发表的关系数据库论文“大型共享数据库数据关系模型”奠定了关系数据库模型的理论基础，使数据库技术成为计算机科学的重要分支。20 世纪 80 年代是关系数据库发展的鼎盛时期，并且至今久盛不衰。关系数据库管理系统的产品如：Oracle, FoxPro, Sybase, Informix, SQL Server 等。

它的最大优点是：

使用了非过程化的数据操纵语言 SQL；