

医学计算机技术基础教程

Visual FoxPro 程序设计

主 编 雷国华 张崇俊

山东科学技术出版社

图书在版编目(CIP)数据

Visual FoxPro 程序设计 /雷国华,张崇俊主编.
—济南:山东科学技术出版社,2006.8
(医学计算机技术基础)
ISBN 7-5331-4370-1

I. V... II. ①雷...②张... III. 关系数据库—
数据库管理系统, Visual FoxPro—程序设计
IV. TP311.138

中国版本图书馆 CIP 数据核字(2006)第 063904 号

医学计算机技术基础教程 Visual FoxPro 程序设计 主编 雷国华 张崇俊

出版者:山东科学技术出版社

地址:济南市玉函路16号
邮编:250002 电话:(0531)82098088
网址:www.lkj.com.cn
电子邮件:sdkj@sdpress.com.cn

发行者:山东科学技术出版社

地址:济南市玉函路16号
邮编:250002 电话:(0531)82098071

印刷者:青岛星球印务有限责任公司

地址:胶南市珠山路120号
邮编:266400 电话:(0532)88194567

开本:787mm×1092mm 1/16

印张:18.25

字数:400千

版次:2006年8月第1版第1次印刷

印数:1-8000

ISBN 7-5331-4370-1 TP·114

定价:25.00元

Visual FoxPro

前言

随着 21 世纪的到来,人类已步入信息社会,信息产业正成为全球经济的主导产业。计算机技术已应用于社会的各个领域,计算机知识水平已经成为素质教育的重要标志。计算机在医学领域中的应用也越来越广泛,对医学技术的发展起到了巨大的推动作用。

关于高校计算机教学的内容,国家教育部提出按三个层次,即计算机文化基础(计算机基础知识、操作系统、网络基础知识、办公自动化软件等)、计算机技术基础(高级语言基础、数据库基础知识等)和计算机应用基础(与专业有关的应用软件)的三层次教学方法,这三个层次之间具有相辅相成的关系。目前,大多数高等院校的学生在大学一年级完成第一层次(计算机文化基础)的学习,大学二年级完成第二层次的学习。第二层次的学习内容,应在一个新的操作系统平台上全面地掌握一种计算机语言的基础知识或数据库基础知识。对医学及其相关专业来说,应结合专业特点有针对性地学习 Visual FoxPro。目前,有关 Visual FoxPro 的教材和参考书较多,门类繁杂,但这些书大部分把 Visual FoxPro 重点放在“可视化”方面。我们认为,重点应放在学习语言的基本概念、数据库中表的基本操作以及程序设计的基本方法上。因为可视化程序设计内容繁杂,就 Visual FoxPro 来讲,一个问题可能用到多个涉及到的控件、属性、类等等,不可能在很短的时间内掌握,此部分内容可让学生自学。医学相关专业学生学习计算机技术基础知识的目的,一是掌握语言的基本使用方法,为以后的应用打基础;二是通过学习来锻炼学生的思维模式。特别是程序设计部分能锻炼学生综合分析问题的能力,使他们养成严谨的思维方式和开阔的思路,增强学生的逻辑判断能力,对培养医学相关专业的学生以严谨的思想面对疾病诊断及预防等都很有帮助。另外,目前众多的 Visual FoxPro 教材和参考书所列实例大都没有与医学专业的相关信息内容结合起来。因此我们编写了本教材,以完成第二层次的计算机教学和学习。

本书以介绍数据库和数据表的基本操作、结构化程序设计和面向对象程序设计的基本方法为重点。内容示例主要以医学专业相关信息和现实生活中的实例为主,以便于医学相关专业学生的学习和理解。学生学完本课程后,可参加全国计算机等级考试(二级 Visual FoxPro)。

本书第一章由束德勤编写,第二章由谢芬编写,第三章由王玉凤编写,第四章由薛慧编写,第五章、第六章、第八章和附录部分由雷国华编写,第七章和第十一章由刘海青编写,第九章和

第十章由杨春波编写,第十二章由徐静编写。雷国华、张崇俊对全书进行了统稿和审校,赵阿妮、王德臣和魏飞对全书应用实例内容进行了上机测试。

本书可作为医学或相关专业学习“计算机技术基础”的教材,也可作为自学“Visual FoxPro程序设计”的参考书,同时还可作为参加全国计算机等级考试(二级 Visual FoxPro)的辅导教材。

由于编写时间仓促,再加上编者水平有限,本书可能存在诸多不足之处,欢迎读者斧正。

编 者

Visual FoxPro

目录

第一章 数据库基础理论	1
1.1 数据库系统概述	1
1.1.1 数据、数据库、数据库管理系统、数据库系统.....	1
1.1.2 数据管理技术的产生和发展	2
1.2 数据模型	5
1.2.1 层次模型	6
1.2.2 网状模型	8
1.2.3 关系模型	9
1.3 关系数据库.....	10
1.3.1 关系数据库的基本概念.....	10
1.3.2 关系运算.....	11
1.4 数据库系统的组成.....	12
1.5 数据库技术的研究领域.....	14
习 题	15
第二章 Visual FoxPro 系统初步	16
2.1 Visual FoxPro 6.0 的主要功能	16
2.1.1 Visual FoxPro 6.0 的主要功能	16
2.1.2 Visual FoxPro 6.0 的主要性能指标	17
2.2 Visual FoxPro 6.0 的安装与启动	18
2.2.1 安装 Visual FoxPro 6.0	18
2.2.2 启动与退出 Visual FoxPro 6.0	19
2.3 Visual FoxPro 6.0 的用户界面	20
2.3.1 Visual FoxPro 的菜单系统与工具栏	21
2.3.2 Visual FoxPro 配置	24
2.4 项目管理器	27
2.4.1 创建项目.....	27

2.4.2	使用项目管理器	31
2.4.3	定制“项目管理器”	33
2.5	Visual FoxPro 向导、设计器、生成器简介	34
2.5.1	Visual FoxPro 的向导	34
2.5.2	Visual FoxPro 的设计器	35
2.5.3	Visual FoxPro 的生成器	35
2.6	Visual FoxPro 6.0 的文件类型	37
2.6.1	数据库文件	37
2.6.2	文档文件	37
2.6.3	程序文件	37
	习 题	38
第三章	数据与数据运算	39
3.1	常量与变量	39
3.1.1	常量	39
3.1.2	变量	40
3.1.3	内存变量的常用命令	41
3.2	Visual FoxPro 中的常用函数	43
3.2.1	数值计算函数	43
3.2.2	字符处理函数	44
3.2.3	日期和时间函数	46
3.2.4	数据类型转换函数	47
3.2.5	测试函数	48
3.3	运算符与表达式	50
3.3.1	字符表达式	50
3.3.2	数值表达式	51
3.3.3	时间日期表达式	51
3.3.4	关系表达式	51
3.3.5	逻辑表达式	52
3.3.6	运算符优先级	53
	习 题	53
第四章	数据库的基本操作	54
4.1	数据库的建立与使用	54
4.1.1	Visual FoxPro 数据库相关概念	54
4.1.2	建立数据库	55
4.1.3	使用数据库	58
4.2	数据库表的建立与基本操作	60
4.2.1	表结构的建立与修改	60
4.2.2	向表中加入记录	66

4.2.3	数据的浏览与编辑	67
4.2.4	记录的显示与输出	72
4.2.5	表的索引	72
4.2.6	排序	75
4.2.7	自由表的操作	76
4.3	多表操作	77
4.3.1	选择工作区	77
4.3.2	建立表间关系	78
习 题		84
第五章	关系数据库标准语言——SQL	85
5.1	SQL 概述	85
5.1.1	SQL 的特点	86
5.1.2	SQL 的功能	87
5.1.3	SQL 语言的基本概念	87
5.2	表数据的定义功能	87
5.2.1	表的定义	87
5.2.2	修改基本表的结构	89
5.2.3	基本表的删除	91
5.3	数据查询功能	91
5.3.1	简单查询	91
5.3.2	连接查询	96
5.3.3	嵌套查询	99
5.3.4	集合查询(集合并运算)	102
5.3.5	设定输出目标	103
5.3.6	其他常用子句	104
5.4	数据更新	104
5.4.1	插入数据	104
5.4.2	修改数据	105
5.4.3	删除数据	106
5.5	视图的定义	106
习 题		108
第六章	Visual Foxpr 结构化程序设计基础	109
6.1	程序与程序文件	109
6.1.1	程序的基本概念	109
6.1.2	Visual FoxPro 程序的建立与运行	109
6.2	文件管理命令和函数	111
6.2.1	文件或路径管理命令	111
6.2.2	文件和路径管理相关函数	113

6.3	程序设计辅助命令	115
6.4	基本的输入和输出命令	118
6.5	程序的基本结构	121
6.5.1	顺序结构程序设计	121
6.5.2	选择结构程序设计	121
6.5.3	循环结构程序设计	126
6.5.4	综合示例	132
6.6	子程序及其调用	135
6.6.1	子程序的调用及形式参数的定义	135
6.6.2	过程、过程文件和过程调用	136
6.6.3	返回主程序语句	138
6.6.4	内存变量的作用域	140
6.7	程序调试	142
6.7.1	调试器环境	142
6.7.2	设置断点	143
6.7.3	调试菜单	145
	习 题	146
第七章	视图与查询	148
7.1	创建视图	148
7.1.1	利用视图设计器创建视图	148
7.1.2	利用视图向导创建视图	155
7.1.3	利用视图操作数据库的优势	161
7.2	利用视图更新数据	161
7.3	创建查询	164
7.3.1	用查询设计器创建查询	164
7.3.2	利用查询向导创建查询	167
7.4	使用查询	172
	习 题	176
第八章	面向对象可视化编程基础	177
8.1	面向对象程序设计中的几个概念	177
8.1.1	对象	177
8.1.2	类的概念	178
8.1.3	属性	180
8.1.4	事件与方法	180
8.2	类的创建	181
8.2.1	创建类的一般方法	181
8.2.2	用菜单方式创建类	182
8.2.3	通过编程定义类	184

8.3	对象的操作	185
8.3.1	由类创建对象	185
8.3.2	设置对象属性	185
8.3.3	调用对象的方法和触发对象的事件	187
8.4	对象设计实例	187
8.4.1	用编程方式设计对象	187
8.4.2	用面向对象的方式设计对象	188
	习 题	190
第九章 表单设计及运行		191
9.1	基本概念	191
9.2	创建表单	191
9.2.1	使用“表单向导”或“表单生成器”创建表单	191
9.2.2	使用“表单设计器”创建表单	195
9.2.3	表单设计工具栏	196
9.3	表单的属性、事件、方法程序和数据环境	196
9.3.1	表单的属性	196
9.3.2	表单的事件	198
9.3.3	表单的方法程序	199
9.3.4	表单的数据环境	200
9.4	在表单中使用控件	201
9.4.1	表单控件工具栏	201
9.4.2	表单及表单对象的命令规则	202
9.4.3	常用特殊对象的引用	202
9.4.4	根据任务选择合适的控件	202
9.4.5	允许用户执行特定的操作	203
9.4.6	显示各种特定信息	205
9.4.7	为用户提供一组预先设定的选择	206
9.4.8	使用微调控件接受给定范围的数值输入	208
9.4.9	接受预先不能确定的输入	208
9.4.10	使用表格容器显示和操作多行数据	211
9.4.11	使用计时器控件在给定的时间间隔执行指定操作	212
9.4.12	控件的布局调整	213
9.4.13	表单的扩展	213
9.5	表单的运行和添加	214
9.5.1	表单的运行	214
9.5.2	表单的添加	215
	习 题	215
第十章 菜单设计		216

10.1	菜单的规划	216
10.2	建立菜单	217
10.3	菜单设置	219
10.3.1	菜单项的提示选项设置	219
10.3.2	菜单的常规选项	220
10.3.3	菜单的“设置”代码	220
10.3.4	菜单的清理代码	221
10.4	菜单项任务的指定	221
10.4.1	使用命令完成任务	221
10.4.2	使用过程完成任务	222
10.5	生成菜单程序	222
10.5.1	菜单的预览	222
10.5.2	生成菜单程序	223
10.5.3	查看和修改菜单源程序代码	223
10.5.4	在应用程序中调用菜单程序	224
10.6	快捷菜单设计	224
	习 题	224
第十一章	创建报表	225
11.1	报表布局	225
11.1.1	设置报表页面与布局	225
11.1.2	设置报表中数据的显示格式	225
11.1.3	添加报表控件	226
11.1.4	常用的报表布局	226
11.2	创建报表	228
11.2.1	利用报表设计器创建报表	228
11.2.2	利用报表向导创建报表	230
11.3	修改报表	233
11.4	使用报表	234
	习 题	234
第十二章	应用程序开发	235
12.1	应用程序开发的基本步骤	235
12.1.1	系统需求分析	235
12.1.2	建立应用程序目录结构	236
12.1.3	数据库设计	236
12.1.4	应用程序设计	236
12.1.5	应用程序连编	237
12.1.6	软件测试	237
12.1.7	发布应用程序	237

12.1.8 系统运行与维护·····	237
12.2 主程序设计·····	237
12.2.1 初始化环境·····	237
12.2.2 显示初始的用户界面·····	237
12.2.3 控制事件循环·····	238
12.2.4 恢复原始的开发环境·····	238
12.2.5 设置主程序为主文件·····	238
12.3 连编应用程序·····	239
12.3.1 设置文件的“排除”与“包含”·····	239
12.3.2 连编项目·····	239
12.3.3 连编应用程序·····	241
12.3.4 连编其他选项·····	241
12.3.5 运行应用程序·····	242
12.4 应用程序生成器·····	242
12.4.1 使用应用程序向导·····	242
12.4.2 应用程序生成器·····	243
习 题·····	247
附录·····	249
附录 1 Visual FoxPro 常用命令 ·····	249
附录 2 Visual FoxPro 函数 ·····	255
附录 3 Visual FoxPro 表单或控件属性 ·····	267
附录 4 Visual FoxPro 对象名称与功能 ·····	271
附录 5 Visual FoxPro 控件名称与功能 ·····	272
附录 6 Visual FoxPro 事件名称与功能 ·····	273
附录 7 Visual FoxPro 方法名称与功能 ·····	275
附录 8 Visual FoxPro 常用文件类型 ·····	276
参考文献·····	278

第一章 数据库基础理论

1.1 数据库系统概述

信息在现代社会中起着越来越重要的作用,信息资源的开发和利用水平已成为衡量一个国家综合国力的重要标志。在计算机应用领域中,数据处理是其主要方面。数据库技术就是作为数据处理中的一门技术而发展起来的。

数据处理是指对各种形式的数据进行收集、存储、加工和传播的一系列活动的总和。其目的之一是从大量的、原始的数据中抽取并推导出对人们有价值的信息,以作为行动和决策的依据;二是为了借助计算机科学地保存和管理复杂的、大量的数据,以使人们能够方便而充分地利用这些信息资源。

数据库技术所研究的问题就是如何科学地组织和存储数据,如何高效地获取和处理数据。近年来,数据库技术和计算机网络技术的相互渗透、相互促进,已成为当今计算机领域发展迅速、应用广泛的两大领域。数据库技术不仅应用于事务处理,并且进一步应用到情报检索、人工智能、专家系统、计算机辅助设计等领域。

数据库是 20 世纪 60 年代末 70 年代初发展起来的一门新技术,它的出现使数据处理进入了一个崭新的时代。它把大量的数据按照一定的结构存储起来,在数据库管理系统的集中管理下,实现数据共享。由于数据库具有数据结构化、数据独立性高、数据共享和易于扩充等特点,因此被广泛地应用于各种管理信息系统中,成为当今信息化社会管理和利用信息资源不可缺少的工具。

在系统地介绍数据库的基本概念之前,这里首先介绍一些数据库最常见的术语和基本概念。

1.1.1 数据、数据库、数据库管理系统、数据库系统

数据、数据库、数据库管理系统和数据库系统是数据库技术密切相关的四个基本概念。

1. 数据(DATA)

数据是数据库中存储的基本对象。数据在大多数人头脑中的第一反应就是数字。其实数字只是最简单的一种数据,是数据的一种传统和狭义的理解。广义的理解,数据的种类很多,文字、图形、图像、声音、病人的档案记录等,这些都是数据。

我们可以对数据做如下定义:描述事物的符号记录称为数据。描述事物的符号可以是数字,也可以是文字、图形、图像、声音、语言等,数据有多种表现形式,它们都可以经过数字化后存入计算机。

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,人们直接用自然语言(如汉语)描述。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。

例如:在病人档案中,如果人们最感兴趣的是病人的姓名、性别、年龄、入院病房、入院时间,那么可以这样描述:李明,男,45,呼吸内科,2006/3/1。

因此,这里的病人记录就是数据。对于上面这条病人记录,了解其含义的人会得到如下信息:李明,男,45岁,入住在呼吸内科病房,入院在时间为2006年3月1日。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。

2. 数据库(DataBase,简称DB)

数据库,顾名思义是存放数据的仓库。只不过这个仓库是在计算机存储设备上,而且数据是按一定的格式存放的。

人们收集并抽取出一个应用所需要的大量数据之后,应将其保存起来,以供进一步加工处理,进一步抽取有用信息。在科学技术飞速发展的今天,人们的视野越来越广,数据量急剧增加。过去人们把数据存放在文件柜里,现在人们借助计算机和数据库技术科学地保存和管理大量的复杂的数据,以便能方便而充分地利用这些宝贵的信息资源。

所谓数据库,是长期储存在计算机内、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。

3. 数据库管理系统(DataBase Management System,简称DBMS)

了解了数据和数据库的概念,下一个问题就是如何科学地组织和存储数据,如何高效地获取和维护数据。完成这个任务的是一个系统软件——数据库管理系统。

数据库管理系统是位于用户与操作系统之间的一层数据管理软件。它的主要功能包括以下几个方面:

(1)数据定义功能:DBMS提供数据定义语言(Data Definition Language,简称DDL),用户通过它可以方便地对数据库中的数据对象进行定义。

(2)数据操纵功能:DBMS还提供数据操纵语言(Data Manipulation Language,简称DML),用户可以使用DML操纵数据实现对数据库的基本操作,如查询、插入、删除和修改等。

(3)数据库的运行管理:数据库在建立、运用和维护时,由数据库管理系统统一管理、统一控制,以保证数据的安全性、完整性、多用户对数据的并发使用及发生故障后的系统恢复。

(4)数据库的建立和维护功能:它包括数据库初始数据的输入、转换功能,数据库的转储、恢复功能,数据库的重组织功能和性能监视、分析功能等。这些功能通常是由一些实用程序完成的。

数据库管理系统是数据库系统的一个重要组成部分。

4. 数据库系统(DataBase System,简称DBS)

数据库系统是指在计算机系统中引入数据库后的系统构成,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。应当指出的是,数据库的建立、使用和维护等工作只靠一个DBMS远远不够,还要有专门的人员来完成,这些人被称为数据库管理员(Data Base Administrator,简称DBA)。

在一般不引起混淆的情况下,常常把数据库系统简称为数据库。

1.1.2 数据管理技术的产生和发展

数据库技术是应数据管理任务的需要而产生的。

数据的处理是指对各种数据进行收集、存储、加工和传播的一系列活动的总和。数据管理

则是指对数据进行分类、组织、编码、存储、检索和维护,它是数据处理的中心问题。人们借助计算机进行数据处理是近 30 年的事。研制计算机的初衷是利用它进行复杂的科学计算。随着计算机技术的发展,其应用远远地超出了这个范围。在应用需求的推动下,在计算机硬件、软件发展的基础上,数据管理技术经历了人工管理、文件系统、数据库系统 3 个阶段。

1. 人工管理阶段

20 世纪 50 年代中期以前,是计算机用于数据处理的初级阶段。在该阶段,应用程序中除了要规定数据的逻辑结构外,还要考虑数据在计算机中如何存储和组织,并为数据分配空间、决定存取方法。数据结构的改变,数据存取的物理地址或存储设备的变化,都会要求修改相应的应用程序。

人工管理数据具有如下的特点:

(1)数据不保存。由于当时计算机主要用于科学计算,一般不需要将数据长期保存,只要在计算某一课题时将数据输入,用完清除。

(2)数据没有相应的软件系统管理。数据由应用程序管理。应用程序既要设计数据的逻辑结构,还要设计物理结构,包括存储结构、存取方法以及输入方式等。

(3)数据不共享。一般一组数据依附于一个应用程序,当多个应用程序设计某些相同数据时,程序与程序之间会有大量的冗余数据。

(4)数据不独立。数据的结构发生变化后,需对应用程序做相应修改。应用程序和数据一一对应,数据和处理它的应用程序混为一个整体。

2. 文件系统管理阶段

20 世纪 50 年代中期到 60 年代中期,计算机软件方面出现了操作系统。在操作系统中,已经有了专门的数据管理软件,一般称为文件系统,对数据的管理有了较大的改进。数据被组织在文件中,可以按名引用,应用程序通过文件管理系统与数据文件发生联系,数据的物理结构和逻辑结构间实现了转换,从而提高了数据的物理独立性。在文件系统中,还提供了多种文件组织形式,如顺序文件组织、索引文件组织和直接存取文件组织等。

在这一阶段,用文件管理数据的优点是数据可以长期保存和由文件系统管理数据等,但仍存在以下缺点:

(1)编程不方便。操作系统只提供低级文件操作命令,对数据的查询、修改、排序等操作必须由编程来解决。

(2)数据冗余量大。为了兼顾各种应用程序的要求,在设计文件系统时,不得不按最大的需求定义数据格式,造成数据冗余。数据冗余不但浪费存储空间,而且会带来数据的不一致。

(3)数据独立性不好。即文件与应用程序间缺乏独立性,文件结构的每一次修改都将导致应用程序的修改。

(4)不支持并发访问。文件系统一般不支持多个应用程序对同一数据文件的并发访问,当一个程序查询某一些数据,而另一个程序正在修改数据时,有可能不一致,甚至错误。

(5)数据缺少统一管理。在数据的结构、编码、表示格式、命名以及输出格式等方面不容易做到规范化、标准化,在数据安全和保密方面也难以采取有效的措施,数据的共享性差、冗余度大以及数据的独立性相对较差。

3. 数据库系统管理阶段

20 世纪 60 年代后期,计算机用于管理的规模越来越大,应用越来越广泛,数据量急剧增大,多种应用、多种语言互相覆盖的共享数据要求越来越强,文件系统处理数据存在着诸多不足,随

着大容量和快速存取的磁盘设备开始进入市场,给数据库系统的研究提供了良好的物质基础。

数据库系统是在文件系统的基础上发展起来的新技术,它克服了文件系统的缺点,解决了冗余和数据依赖问题,提供了更广泛的数据共享,为应用程序提供了更高的独立性,保证了数据的完整性和安全性,并为用户提供了方便的用户接口。从文件系统到数据库系统,标志着数据库管理技术的飞跃。下面给大家介绍数据库系统的特点及由此带来的优点。

与人工管理和文件系统相比,数据库系统的特点主要有以下几个方面:

(1)数据结构化:数据结构化是数据库与文件系统的根本区别。在文件系统中,相互独立的文件的记录内部是有结构的。传统文件的最简单形式是等长同格式的记录集合。

数据库系统中的数据是相互关联的,这种关联不仅表现在记录内部,更重要的是记录类型之间的相互联系。整个数据库是以一定的形式构成的。

(2)数据的共享性高,冗余度低,易扩充:数据库系统从整体角度看待和描述数据,数据不再面向某个应用而是面向整个系统,因此数据可以被多个用户、多个应用共享使用。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与不一致性。

所谓数据的不一致性,是指同一数据不同拷贝的值不一样。采用人工管理或文件系统管理时,由于数据被重复存储,当不同的应用使用和修改不同的拷贝时,就很容易造成数据的不一致。在数据库中数据共享,减少了由于数据冗余造成的一致现象。

由于数据面向整个系统,是有结构的数据,不仅可以被多个应用共享使用,而且容易增加新的应用,这就使得数据库系统弹性大,易于扩充,可以适应各种用户的要求。可以取整体数据的各种子集用于不同的应用系统,当应用需求改变或增加时,只要重新选取不同的子集或加上一部分数据便可以满足新的需求。

(3)数据独立性高:数据独立性是数据库领域中的一个常用术语,包括数据的物理独立性和数据的逻辑独立性。

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怎样存储是由 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,这样当数据的物理存储改变了,应用程序不用改变。

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,用户程序也可以不变。

数据与程序的独立,把数据的定义从程序中分离出去,加上数据的存取又由 DBMS 负责,从而简化了应用程序的编制,大大减少了应用程序的维护和修改。

(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同一个数据。为此,DBMS 还必须提供以下几方面的数据控制功能:

①数据的安全性(Security)保护:数据的安全性是指保护数据,以防止不合法使用造成数据的泄密和破坏。使每个用户只能按规定对某些数据以某些方式进行使用和处理。

②数据的完整性(Integrity)检查:数据的完整性是指数据的正确性、有效性和相容性。完整性检查将数据控制在有效的范围内,或保证数据之间满足一定的关系。

③并发(Concurrency)控制:当多个用户的并发进程同时存取、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,因此必须对多用户的并发操作加以控制和协调。

④数据库恢复(Recovery):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,也会影响数据库中数据的正确性,甚至造成数据库部分或全部数据的丢失。DBMS 必须具有将数据库从错误状态恢复到某一已知的正确状态(也称为完整状态或一致状态)的功能,这就是数据库的恢复功能。

综上所述,数据库是长期存储在计算机内有组织的、大量的、共享的数据集合。它可以供各种用户共享,具有最小冗余度和较高的数据独立性。DBMS 在数据库建立、运用和维护时,对数据库进行统一控制,以保证数据的完整性、安全性,并在多用户同时使用数据库时进行并发控制,在发生故障后对系统进行恢复。

下面以表格的形式列出数据管理技术 3 个阶段的比较(表 1-1)。

表 1-1 数据管理 3 个阶段的比较

管理阶段		人工管理阶段	文件系统阶段	数据库系统阶段
背景特点	应用背景	科学计算	科学计算、管理	大规模管理
	硬件背景	无直接存取存储设备	磁盘、磁鼓	大容量磁盘
	软件背景	无操作系统	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	数据的管理者	用户(程序员)	文件系统	数据库管理系统
	数据面向的对象	某一应用程序	某一应用	现实世界
	数据的共享程度	无共享、冗余度极大	共享性差、冗余度大	共享性高、冗余度小
	数据的独立性	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
	数据的结构化	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	由数据库管理系统提供数据安全、完整性、并发控制和恢复能力

数据库系统的出现,使信息系统从以加工数据的程序为中心转向围绕共享的数据库为中心的新阶段。这样既便于数据的集中管理,又有利于应用程序的研制和维护,提高了数据的利用率和相容性,提高了决策的可靠性。

目前,数据库已经成为现代信息系统不可分离的重要组成部分。具有数百万甚至数 10 亿字节信息的数据库已经普遍存在于科学技术、工业、农业、商业、服务业和政府部门的信息系统,数据库技术得到广泛的应用和普及。

数据库技术是计算机领域中发展最快的技术之一。数据库技术的发展是沿着数据模型的主线展开的。下面讨论数据模型。

1.2 数据模型

模型,特别是具体模型,人们并不陌生。一张地图、一组建筑设计沙盘、一架精致的航模飞机都是具体的模型。一眼望去,就会使人联想到真实生活中的事物。模型是现实世界特征的模拟和抽象。数据模型(Data Model)也是一种模型,它是现实世界数据特征的抽象。

数据库是某个企业、组织或部门所涉及的数据的综合,它不仅要反映数据本身的内容,而

且要反映数据之间的联系。由于计算机不可能直接处理现实世界中的具体事物,所以人们必须事先把具体事物转换成计算机能够处理的数据。在数据库中,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。

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。

数据模型应满足三方面要求:一是能比较真实地模拟现实世界;二是容易为人所理解;三是便于在计算机上实现。一种数据模型要很好地满足这三方面的要求在目前还很困难。在数据库系统中针对不同的使用对象和应用目的,采用不同的数据模型。

数据模型是数据库系统的核心和基础。各种机器上实现的 DBMS 软件都是基于某种数据模型的。

本节简要介绍层次模型、网状模型和关系模型。

1.2.1 层次模型

层次模型是数据库系统中最早出现的数据模型,层次数据库系统采用层次模型作为数据的组织方式。层次数据库系统的典型代表是 IBM 的 IMS(Information Management System)数据库管理系统,这是 1968 年 IBM 推出的第一个大型的商用数据库管理系统,曾经得到广泛的使用。

层次模型用树形结构来表示各类实体以及实体间的联系。现实世界中,许多实体之间的联系本来就呈现出一种很自然的层次关系,如行政机构、家族关系等。

1. 层次数据模型的数据结构

层次模型的特点是将数据组织成有向有序的树结构。层次模型由处于不同层次的各个结点组成。除根结点外,其余各结点有且仅有一个上一层结点作为其“双亲”,而位于其下的较低一层的若干个结点作为其“子女”。结构中的结点代表数据记录,连线描述位于不同结点数据间的从属关系(限定为“一对多”的关系)。

若用图来表示,层次模型是一棵倒立的树。在数据库中,满足以下条件的数据模型称为层次模型:

- (1)有且只有一个结点,没有“双亲”结点,这个结点称为根结点。
- (2)根以外的其他结点有且只有一个“双亲”结点。

以下就是一个层次模型的实例(图 1-1)。

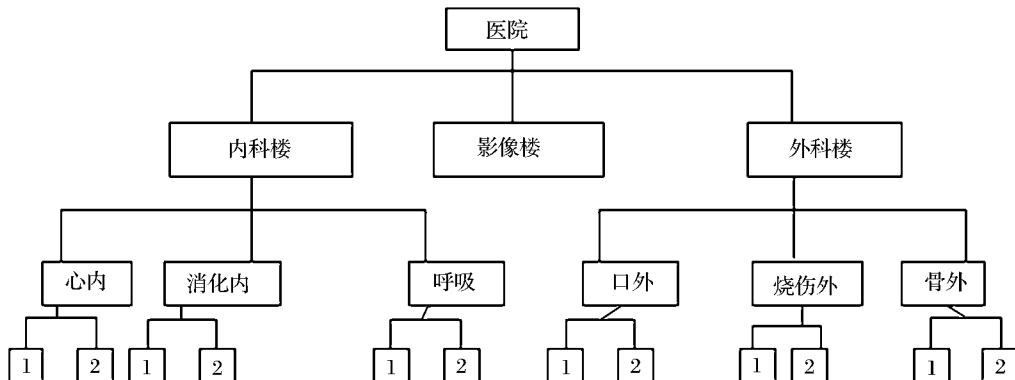


图 1-1