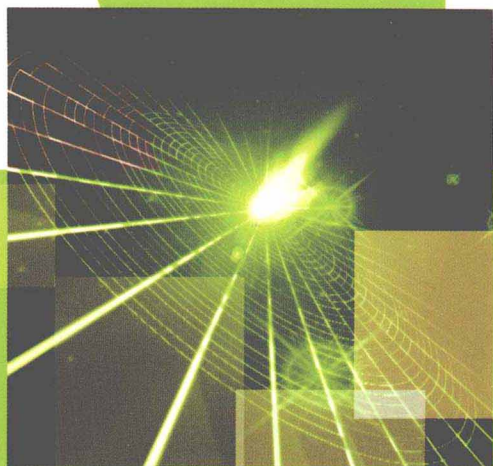




21世纪高等学校计算机科学与技术规划教材



Visual FoxPro
Chengxu Sheji Jichu Jiaocheng

Visual FoxPro

程序设计基础教程

主 编 李曼青 赵庆展 胡 俊
主 审 梁 斌



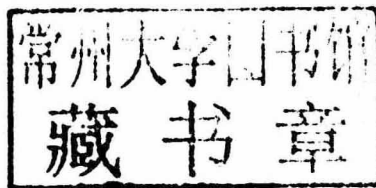
北京邮电大学出版社
www.buptpress.com



21 世纪高等学校计算机科学与技术规划教材

Visual FoxPro 程序设计基础教程

主 编 李曼青 赵庆展 胡 俊
主 审 梁 斌



北京邮电大学出版社
• 北京 •

内容简介

本书全面介绍了 Visual FoxPro 数据库编程语言,内容涵盖了从数据库的基本概念和基本操作到应用程序设计的各个方面。全书的主要内容有:数据库的基础知识、Visual FoxPro 6.0 概述、数据与数据运算、项目管理器、表的基本操作、数据库及其操作、SQL 结构化查询语言、视图与查询、结构化程序设计、表单、报表和标签、菜单设计与应用。书中给出了大量例题,便于初学者入门。

本书可作为高等院校非计算机专业 Visual FoxPro 课程入门教材,也可用于自学。

图书在版编目(CIP)数据

Visual FoxPro 程序设计基础教程/李曼青,赵庆展,胡俊主编. -- 北京:北京邮电大学出版社,2011.1
ISBN 978-7-5635-2530-0

I. ①V… II. ①李… ②赵… ③胡… III. ①关系数据库—数据库管理系统, Visual FoxPro—程序设计—高等学校—教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2010)第 255175 号

书 名	Visual FoxPro 程序设计基础教程
主 编	李曼青 赵庆展 胡 俊
责任编辑	张雪祥
出版发行	北京邮电大学出版社
社 址	北京市海淀区西土城路 10 号(100876)
电话传真	010-82333010 62282185(发行部) 010-82333009 62283578(传真)
电子信箱	ctrd@buptpress.com
经 销	各地新华书店
印 刷	北京市梦宇印务有限公司
开 本	787 mm×1 092 mm 1/16
印 张	16
字 数	365 千字
版 次	2011 年 1 月第 1 版 2011 年 1 月第 1 次印刷

ISBN 978-7-5635-2530-0

定价: 30.00 元

如有质量问题请与发行部联系
版权所有 侵权必究

编 委 会

主 任	梁 斌	
副主任	裘祖旗	胡 俊
委 员	赵庆展	郭 理
	田 敏	陈卫东
	杜文军	李曼青

前 言

Visual FoxPro 6.0 是美国 Microsoft 公司推出的程序开发工具软件,主要用于小型数据库系统的开发。本软件系统不仅可以简化数据管理,使应用程序的开发流程更为合理,而且可以使组织数据、定义数据库规则和建立应用程序等工作变得简单易行。利用可视化的设计工具和向导,可以快速创建表单、查询和报表。Visual FoxPro 6.0 提供了一个集成化的开发环境,不仅拥有功能强大的面向对象程序设计工具以及客户/服务器系统开发能力,而且支持 ActiveX,这些特点为快速开发功能全面的应用程序创造了良好的条件,是目前各类学校学习数据库管理系统开发工具的首选。

从教材科学性、理论性和先进性方面来考虑,本书安排了数据库基础知识、Visual FoxPro 6.0 基本操作、结构化程序设计、表单、报表、菜单等内容。从学生的接受能力及教学方法来考虑,本书在编写手法上按照循序渐进、由浅入深、举一反三、理论与实践相结合的规律出发,以 4~6 个课时为一讲,每一讲有理论知识、例题、思考与练习等内容,全书理论与实践操作环环紧扣,既便于教师教学,又便于学生学习。

全书可按 56~72 课时讲授,理论课与上机操作比例为 1:1,有条件的话可适当增加上机时间,教师也可根据自己的教学经验和学生的实际情况,适当改变章节的顺序或筛选某些内容进行讲解。本书有配套教材《Visual FoxPro 程序设计习题集与上机辅导》,其中提供了习题及答案、上机实验及应用程序实例等内容,与本书配套使用将使学习效果更佳。

本书由李曼青、赵庆展、胡俊担任主编,并负责全书统稿、修改和定编。具体编写情况如下:第 1、第 7 章由刘春编写,第 2、第 3 章由朱东芹编写,第 4、第 5 章由康娟编写,第 6、第 8 章由刘波编写,第 9、第 10 章由李曼青编写,第 11、第 12 章由宁川编写。本书可作为高等院校本、专科学生的教科书,同时也适合具有同等文化程度的读者自学之用。

由于时间仓促和作者水平有限,书中不当之处在所难免,敬请读者批评指正。

编 者

目 录

第 1 章 数据库基础知识	1
1.1 数据、信息与数据处理	1
1.1.1 数据与信息	1
1.1.2 数据处理	2
1.1.3 数据管理技术的发展	2
1.1.4 数据库系统新技术	5
1.2 数据库系统	7
1.2.1 数据库	7
1.2.2 软件系统	7
1.2.3 硬件系统	8
1.2.4 人员	8
1.3 数据模型	9
1.3.1 基本概念	9
1.3.2 实体之间的联系	10
1.3.3 数据模型	10
1.4 关系数据库	13
1.4.1 基本概念	13
1.4.2 关系的完整性约束	15
1.4.3 关系运算	16
思考与练习	16
第 2 章 Visual FoxPro 6.0 概述	17
2.1 Visual FoxPro 的发展	17
2.2 Visual FoxPro 6.0 的安装	17
2.2.1 Visual FoxPro 6.0 的软件、硬件环境	18
2.2.2 Visual FoxPro 6.0 的安装过程	18
2.2.3 Visual FoxPro 6.0 的主要特点	21
2.3 Visual FoxPro 6.0 的用户界面	22
2.3.1 Visual FoxPro 6.0 的启动	22
2.3.2 Visual FoxPro 6.0 的退出	24
2.3.3 Visual FoxPro 6.0 的用户界面	24
2.3.4 Visual FoxPro 6.0 的菜单系统	25
2.4 Visual FoxPro 6.0 的工作方式	25

2.4.1	Visual FoxPro 6.0 命令的格式与特点	25
2.4.2	两类工作方式	26
2.5	Visual FoxPro 6.0 的辅助设计工具	27
2.5.1	向导	27
2.5.2	设计器	28
2.5.3	生成器	29
	思考与练习	30
第3章	数据与数据运算	31
3.1	数据和数据类型	31
3.2	常量与变量	32
3.2.1	常量	32
3.2.2	变量	34
3.3	表达式	37
3.3.1	算术运算符及表达式	37
3.3.2	字符运算符及表达式	38
3.3.3	日期时间运算符及表达式	38
3.3.4	关系运算符及表达式	39
3.3.5	逻辑运算符及表达式	40
3.4	常用函数	41
3.4.1	数值函数	41
3.4.2	字符函数	42
3.4.3	日期函数	44
3.4.4	数据类型转换函数	44
3.4.5	测试函数	46
	思考与练习	47
第4章	项目管理器	49
4.1	项目管理器的创建	49
4.1.1	利用菜单生成项目文件	49
4.1.2	利用命令生成项目文件	50
4.2	项目管理器的使用	50
4.2.1	“项目管理器”的选项卡	50
4.2.2	“项目管理器”的按钮	51
4.2.3	“项目管理器”的使用	51
4.3	项目管理器的定制	55
4.3.1	移动	55
4.3.2	缩放	55
4.3.3	折叠	55
4.3.4	拆分	55
4.3.5	停放	56

思考与练习	57
第 5 章 表的基本操作	58
5.1 表的建立	58
5.1.1 表的基本概念	58
5.1.2 表结构的建立	60
5.1.3 表中记录的输入	62
5.2 表的基本操作	64
5.2.1 表的打开与关闭	64
5.2.2 表的显示	66
5.2.3 表的修改	67
5.2.4 记录指针的定位	72
5.2.5 表中记录的增删	73
5.2.6 数据的统计与计算	75
5.2.7 表的复制	77
5.3 索引	77
5.3.1 排序	77
5.3.2 索引的基本概念	78
5.3.3 索引的建立	80
5.3.4 索引的使用	85
5.4 多表操作	88
5.4.1 多工作区的概念	88
5.4.2 工作区的选择与互访	88
5.4.3 表的关联	89
5.4.4 表间的连接	92
思考与练习	94
第 6 章 数据库及其操作	95
6.1 建立和打开数据库	95
6.1.1 建立数据库	95
6.1.2 打开、关闭、删除数据库	96
6.2 数据库中表的基本操作	98
6.2.1 新建表	98
6.2.2 添加表	99
6.2.3 移去和删除表	101
6.3 设置数据库表的属性	101
6.3.1 设置表中字段的显示标题	101
6.3.2 设置表中字段的输入输出掩码	103
6.3.3 设置表中字段的默认值	105
6.3.4 设置表中字段的有效性规则	106
6.3.5 设置触发器	107

6.4 建立永久关系	109
6.4.1 建立表间的永久关系	109
6.4.2 设置参照完整性	110
思考与练习	113
第7章 SQL 结构化查询语言	114
7.1 SQL 语言概述	114
7.2 数据定义	115
7.2.1 创建表结构	115
7.2.2 修改表的结构	116
7.2.3 删除数据表	118
7.3 数据操纵	118
7.3.1 插入记录	118
7.3.2 更新记录	119
7.3.3 删除记录	119
7.4 数据查询	120
7.4.1 基本查询	120
7.4.2 条件查询	121
7.4.3 多表查询	123
7.4.4 连接查询	124
7.4.5 分组查询	124
7.4.6 嵌套查询	125
7.4.7 查询结果处理	126
思考与练习	128
第8章 视图与查询	129
8.1 视图	129
8.1.1 视图的概念	129
8.1.2 视图的创建和使用	129
8.2 查询的创建与使用	138
8.2.1 查询的概念	138
8.2.2 查询的创建和使用	138
思考与练习	145
第9章 结构化程序设计	146
9.1 结构化程序设计思想	146
9.1.1 结构化程序设计	146
9.1.2 结构化程序设计的优点	146
9.1.3 程序设计基础	146
9.2 程序文件编辑器	148
9.2.1 用菜单方式进入	148
9.2.2 用命令方式进入	148

9.2.3 程序的建立与执行	149
9.2.4 简单程序设计	150
9.3 顺序结构程序	151
9.3.1 程序中的辅助命令	151
9.3.2 交互式输入命令	152
9.3.3 格式输入输出命令	153
9.4 选择结构程序	153
9.4.1 简单分支选择语句	153
9.4.2 双分支选择语句	154
9.4.3 IF 语句的嵌套	155
9.4.4 多分支选择语句	156
9.5 循环结构程序	158
9.5.1 DO WHILE 循环	158
9.5.2 FOR 循环	160
9.5.3 SCAN 循环	162
9.5.4 循环嵌套	163
9.6 程序的模块化	164
9.6.1 子程序的结构	164
9.6.2 应用举例	166
思考与练习	167
第 10 章 表单设计与应用	168
10.1 简单表单的创建	168
10.1.1 表单创建的方法与种类	168
10.1.2 利用向导创建表单	168
10.1.3 保存及运行表单	172
10.1.4 创建一对多表单	172
10.2 快速创建表单及运行	176
10.2.1 启动表单设计器	176
10.2.2 快速创建表单	178
10.2.3 利用命令创建表单	180
10.2.4 表单的运行	181
10.3 表单的操作	182
10.3.1 表单控件简介	182
10.3.2 表单的属性	182
10.3.3 表单的常用事件与方法	183
10.4 常用表单控件	184
10.4.1 文本框、标签控件	184
10.4.2 命令按钮和命令按钮组控件	185
10.4.3 选项按钮组和复选框控件	187

10.4.4 列表框和组合框控件	190
10.4.5 表格和页框控件	193
10.4.6 计时器和微调控件	194
思考与练习	197
第 11 章 报表和标签设计与应用	199
11.1 报表概述	199
11.1.1 报表的组成	199
11.1.2 报表的类型	199
11.1.3 报表的常规布局	200
11.1.4 创建报表的步骤	201
11.1.5 创建报表的方法	202
11.2 创建简单报表	202
11.2.1 创建单表报表	202
11.2.2 创建一对多报表	205
11.2.3 快速报表	208
11.3 报表设计器	211
11.3.1 报表设计器界面	211
11.3.2 报表设计器工具栏	213
11.3.3 报表的数据环境	219
11.3.4 分组报表	219
11.4 报表的打印输出	223
11.4.1 使用菜单输出报表	223
11.4.2 使用命令输出报表	224
11.5 标签	224
思考与练习	227
第 12 章 菜单设计与应用	228
12.1 菜单概述	228
12.1.1 菜单的结构及类型	228
12.1.2 建立菜单系统的步骤	229
12.1.3 系统菜单的控制	230
12.2 菜单设计器	231
12.2.1 菜单设计器界面	231
12.2.2 菜单“常规选项”和“菜单选项”	234
12.2.3 创建快速菜单	239
12.2.4 创建顶层菜单	239
12.3 快捷菜单	242
思考与练习	245
参考文献	246

第 1 章 数据库基础知识

数据库系统(Database System,DBS)是指引进数据库技术的计算机系统。从 20 世纪 60 年代末起,数据库技术作为数据处理的一门新技术开始得到大力发展。数据库技术是计算机科学技术的一个重要分支,也是计算机科学技术中发展最快的领域之一,经过 40 年的发展已形成了较为完整的理论体系。数据库已经成为各领域存储数据、管理信息、共享资源的基本手段。作为学习的理论先导,本章介绍数据库系统基础知识。

1.1 数据、信息与数据处理

1.1.1 数据与信息

数据和信息是数据处理中的两个基本概念,数据与信息有着不同的含义,它们既有联系又有区别。

1. 数据

(1)数据的定义

数据是用来描述客观事物的可识别的符号。例如,“2010070123”,“张小宁”,“外语”,“170”就是数据。它是数据库存储和处理的基本对象。

(2)数据的表现形式

数据在大多数人们头脑中的第一个反应就是由 0~9 十个数码组成的数据,其实这是一种传统的和狭义的理解。广义地理解,描述事物的符号可以是数字,也可以是文字、声音、图形、图像等。数据可以有多种表现形式,它们都可以经过数字化后存入计算机。

2. 信息

(1)信息的定义

数据的表现形式还不能完全表达其内容,需要通过解释和处理。只有给数据赋予确切的含义后,它对人们才是有用的。例如,描述某学生的数据可解释为:张小宁是外语系的学生,学号是 2010070123,身高是 170 cm。人们通过各种方式所获得的数据,利用解释、分析、推理、归纳和综合等方法,从中获得有用的数据称为信息。因此,简单地说,数据是信息存在的一种形式,只有通过解释或处理之后,有使用价值的数据才能成为信息。

(2)信息的特征

信息是客观事物运动状态和存在方式的反映,具有如下主要特征。

①信息源于物质和能量。信息不可能脱离物质而存在,信息的传递需要物质载体,信息的

获取和传递要消耗能量。任何信息都不能脱离具体的符号及其物质载体而单独存在,所以说,没有物质载体,信息就不能存储和传递,如信息可以通过网络、电视、广播、报刊、杂志等进行传递。

②信息是可以感知的。人类对客观事物的感知,可以通过感觉器官,也可以借助各种仪器设备感知。不同的信息源有不同的感知形式,如报纸上刊登的信息通过视觉器官感知,广播中的信息通过听觉器官感知。

③信息是可存储、传递、加工和再生的。人类除了运用大脑记忆信息外,还可以运用语言、文字、图像、符号等记载信息。如果要使信息长期保存下来,还必须借助纸张、磁盘等物质作为信息的载体加以存储,再通过电视、广播、计算机网络等信息媒体进行传播。信息是可以加工的,它可以被压缩、存储、有序化,也可以转换形态。

④信息是有用的。它是人们活动所必需的知识,利用信息能够克服工作中的盲目,增加主动性和科学性,可以把事情做得更好。

3. 数据与信息的关系

信息和数据既有联系,又有区别。数据是信息的符号表示或载体,信息则是数据的内涵,是对数据的语义解释;数据是物理性的,是被加工的对象,而信息是对数据加工的结果,是观念性的,并依赖于数据而存在。数据表示了信息,而信息只有通过数据形式表示出来才能被人们理解和接受。信息是有用的数据,信息是通过数据符号来传播的,数据如不具有知识性和有用性则不能称其为信息。尽管两者在概念上不尽相同,但通常人们并不严格去区分它们。

1.1.2 数据处理

数据处理是指将数据转换成信息的过程。包括对数据进行收集、整理、存储、加工、检索和传输等一系列活动,其目的是从大量的、杂乱无章的数据中整理出对人们有价值、有意义的信息(即信息),作为决策的依据。所以,数据处理也称为信息处理。信息处理的真正含义是为了产生信息而处理数据。

1.1.3 数据管理技术的发展

数据管理指的是对数据的分类、组织、编码、存储、检索和维护等。它是数据处理的核心。数据管理技术的优劣,将直接影响数据处理的效率。到目前为止,数据管理大致经历了三个阶段:人工管理阶段、文件系统阶段、数据库系统阶段。

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。外部存储器只有磁带、卡片和纸带,软件只有汇编语言,尚无数据管理方面的软件。数据处理的方式基本上是批处理。这个时期数据管理的特点如下。

①数据不保存。因为当时计算机主要用于科学计算,数据量少,对于数据保存的需求尚不迫切。需要时把数据输入内存,运算后将结果输出。数据不保存在计算机中。

②没有专用的软件对数据进行管理。在应用程序中,不仅要管理数据的逻辑结构,还要设计其物理结构、存取方法、输入输出方法等。当存储改变时,应用程序中存取数据的子程序就

需随之改变。

③数据不具有独立性。数据的独立性是指数据和程序之间的依赖程度,包括逻辑独立性和物理独立性。当数据的类型、格式或输入输出方式等逻辑结构或物理结构发生变化时,必须对应用程序作出相应的修改,应用程序和数据相互依赖,即数据不具有独立性。

④数据是面向程序的。一组数据只对应于一个应用程序。即使两个应用程序都涉及某些相同数据,也必须各自定义,无法相互利用。因此,在程序之间有大量的冗余数据。

在人工管理阶段,用户、数据与程序的关系如图 1-1 所示。

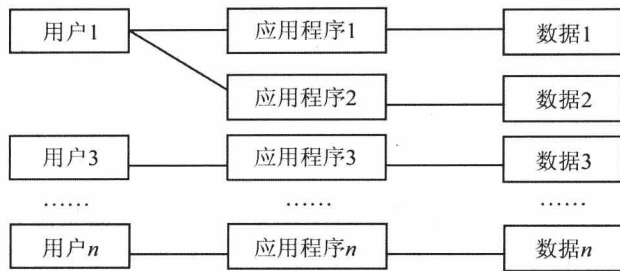


图 1-1 人工管理阶段用户、数据与程序的关系

2. 文件系统阶段

20 世纪 50 年代后期到 60 年代中期,计算机不仅用于科学计算,还用于信息管理。此时,外部存储器已有磁盘、磁鼓等直接存取的存储设备;软件领域出现了高级语言和操作系统。操作系统中的文件系统是专门的数据管理软件。这时可以把相关的数据组成一个文件存放在计算机中,在需要时只要提供文件名,计算机就能从文件系统中找出所要的文件,把文件中存储的数据提供给用户进行处理。

文件系统管理阶段有以下几个特点。

①数据以“文件”形式可长期保存在外部存储器的磁盘上。应用程序可对文件进行大量的检索、修改、插入和删除等操作。

②文件组织已多样化。有索引文件、顺序存取文件和直接存取文件等。因而对文件中的记录可顺序访问,也可随机访问,便于存储和查找数据。

③数据与程序间有一定的独立性。数据由专门的软件即文件系统进行管理,程序和数据间由软件提供的存取方法进行转换,数据存储发生变化不一定影响程序的运行。

④对数据的操作以记录为单位。这是由于文件中只存储数据,不存储文件记录的结构描述信息。文件的建立、存取、查询、插入、删除、修改等所有操作,都要用程序来实现。

在文件系统阶段,用户虽有了一定的方便,但仍有很多缺点。主要表现在以下几方面。

①数据冗余度大。由于各数据文件之间缺乏有机的联系,造成每个应用程序都有对应的文件,有可能同样的数据在多个文件中重复存储,数据不能共享。

②数据独立性低。数据和程序仍相互依赖,一旦改变数据的逻辑结构,必须修改相应的应用程序。而应用程序发生变化,如改用另一种程序设计语言来编写程序,也需修改数据结构。

③数据一致性差。由于相同数据的重复存储、各自管理,在进行更新操作时,容易造成数据的不一致。

这样,文件系统仍然是一个不具有弹性的无结构的数据集合。文件之间是孤立的,不能反

映现实世界中事物之间的内在联系。在文件系统阶段,用户、数据与程序的关系如图 1-2 所示。

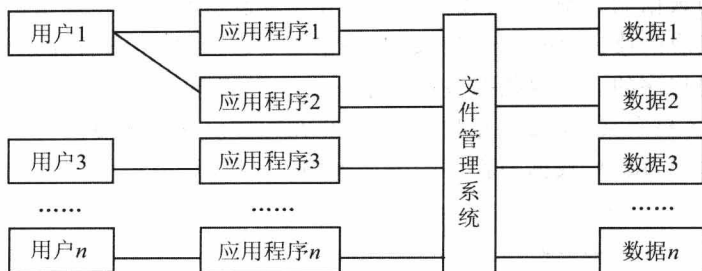


图 1-2 文件系统阶段用户、数据与程序的关系

3. 数据库系统阶段

20 世纪 60 年代末,计算机应用于管理的规模更加庞大,数据量急剧增加,数据共享性更强;硬件方面出现了大容量磁盘,使计算机联机存取大量数据成为可能;硬件价格下降,而软件价格上升,使开发和维护系统软件的成本增加。文件系统的数据管理方法已无法适应开发应用系统的需要。为解决多用户、多个应用程序共享数据的需求,出现了统一管理数据的专门软件系统,即数据库管理系统,这使利用数据库技术管理数据变成了现实。

数据库系统阶段有以下几个特点。

①数据共享性高、冗余度低。这是数据库系统阶段的重大改进,数据不再面向某个应用程序而是面向整个系统,当前所有用户可同时访问数据库中的数据。这样便减少了不必要的数据冗余,节约了存储空间,同时也避免了数据之间的不相容性与不一致性。

②数据结构化。即按照某种数据模型,将应用的各种数据组织到一个结构化的数据库中。在数据库中数据的结构化,不仅要考虑某个应用的数据结构,还要考虑整个系统的数据结构,并且还要能够表示出数据之间的有机关联。

③数据独立性高。数据的独立性是指逻辑独立性和物理独立性。

数据的逻辑独立性是指当数据的总体逻辑结构改变时,数据的局部逻辑结构不变。由于应用程序是依据数据的局部逻辑结构编写的,所以应用程序不必修改,从而保证了数据与程序间的逻辑独立性。

数据的物理独立性是指当数据的存储结构改变时,数据的逻辑结构不变,从而应用程序也不必改变。

④有统一的数据控制功能。数据库为多个用户和应用程序所共享,对数据的存取往往是并发的,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。为确保数据库数据的正确有效和数据库系统的有效运行,数据库管理系统提供下述四方面的数据控制功能。

- 数据的安全性控制:防止不合法使用数据造成数据的泄露和破坏,保证数据的安全和机密。例如,系统提供口令检查或其他手段来验证用户身份,防止非法用户使用系统;也可以对数据的存取权限进行限制,只有通过检查后才能执行相应的操作。

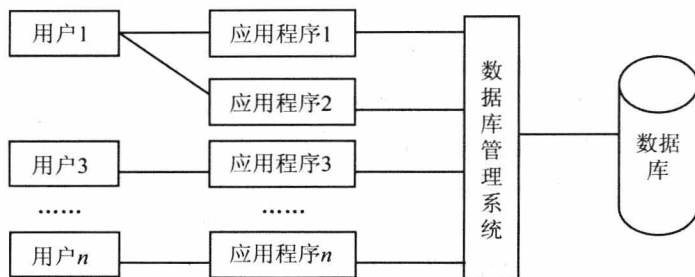
- 数据的完整性控制:系统通过设置一些完整性规则以确保数据的正确性、有效性和相容性。正确性是指数据的合法性,如年龄属于数值型数据,只能包含 0,1,...,9,不能包含字母

或特殊符号。有效性是指数据是否在其定义的有效范围内,如月份只能用1~12之间的正整数表示。相容性是指表示同一事实的两个数据应相同,否则就不相容,如一个人不能有两个性别。

● 并发控制:防止多用户同时存取或修改数据库时,因相互干扰而提供给用户不正确的数据,并使数据库受到破坏。

● 数据恢复:当数据库被破坏或数据不可靠时,系统有能力将数据库从错误状态恢复到最近某一时刻的正确状态。

数据库系统阶段,用户、数据与程序之间的关系可用图1-3表示。



1.1.4 数据库系统新技术

目前,世界上已有许多大型数据库系统在运行,其应用已深入到人类社会生活的各个领域,并在计算机网络的基础上,建立了许多国际性的联机检索系统。由于传统的数据库系统已无法满足各种应用的需求,因此从20世纪80年代开始数据库研究出现了许多新的领域,相继研究出了分布式数据库系统、面向对象数据库系统和网络数据库系统等。

1. 分布式数据库系统

分布式数据库是一个物理上分布在计算机网络的不同结点,但在逻辑上又同属于一个系统的数据集合。在分布式数据库系统中,数据库存储在几台计算机中,这几台计算机之间通过高速网络相互通信,系统中每一台计算机被称为一个结点。

在分布式数据库系统中,应用分为局部应用和全局应用两种。局部应用是指仅操作本地结点上数据库的应用;而全局应用是指需要操作两个或两个以上结点中的数据库的应用。例如,一个银行系统中,有多个分支机构分布在不同的城市。每个分支机构有自己的服务器(结点),用来维护该分支机构的所有账户的数据库。同时有若干客户机,用于完成本地客户的存、取款业务等(局部应用)。分支机构的客户机也可以完成某些全局应用,如不同分支机构中账户之间的转账,就需要同时访问和更新两个结点上的数据库中的数据。不支持全局应用的系统不能称为分布式数据库系统。同时,分布式数据库系统不仅要求数据的物理分布,而且要求这种分布是面向处理、面向应用的。

分布式数据库系统是物理上分散,逻辑上集中的数据库系统。系统中的数据分别存放在计算机网络的不同结点上,网络中的每个结点具有独立处理的能力(称为场地自治),可以执行局部应用,每个结点也可以通过网络通信子系统执行全局应用。

分布式数据库系统主要有如下几个特点。

①数据的物理分布性。数据库中的数据分布在计算机网络的不同结点上,而不是集中在一个结点上。因此,它不同于通过计算机网络共享的集中式数据库系统。

②数据的逻辑整体性。分布在计算机网络不同结点上的数据在逻辑上属于同一个系统,因此,它们在逻辑上是相互联系的整体。

③结点的自主性。每个结点有自己的计算机,自己的数据库(即局部数据库,简称 LDB),自己的数据库管理系统(LDBMS),因而能独立地管理局部数据库。局部数据库中的数据可以供本结点的用户存取(局部应用),也可以供其他结点上的用户存取以供全局应用。

2. 面向对象数据库系统

面向对象数据库系统(Object Oriented DataBase System, OODBS)是将面向对象的模型、方法和机制,与先进的数据库技术有机地结合而形成的新型数据库系统。面向对象数据库系统的研究有两种观点:一种是在面向对象程序设计语言中引入数据库技术,使应用程序能够存取并处理对象;另一种是扩展数据库系统,使其具有面向对象的特征。现在一般把前一类数据库系统称为面向对象数据库系统,后一类称为对象关系数据库系统(Object Relation DataBase System, ORDBS),这两类统称为对象数据库系统。

目前,各个关系数据库厂商都在不同程度上扩展了关系模型,推出了符合面向对象数据模型的数据库系统。面向对象的数据模型吸收了面向对象程序设计方法的核心概念和基本思想,用面向对象的观点来描述现实世界的实体。

对象关系数据库系统可定义为:在关系数据模型的基础上,提供元组、数组、集合等丰富的数据类型以及处理新的数据类型的能力,并且具有继承性和对象标识等面向对象特点,这样形成的数据模型称为对象关系数据模型。基于对象关系模型的数据库系统称为对象关系数据库系统。对象关系数据库系统除了具有原来关系数据库的各种特点外,还具有以下特点。

①扩充数据类型。以关系数据库和 SQL 为基础,扩展关系数据模型,增加面向对象的数据类型和特性。新的数据类型可定义为原有类型的子类或超类。新的数据类型定义之后,存放在数据库管理系统中,如同基本数据类型一样,可供所有用户共享。

②支持复杂对象。OODBS 中的基本结构是对象而不是记录,一个对象不仅包括描述它的数据,还包括对它操作的方法。它不仅支持简单的对象,还支持由多种基本数据类型或用户自定义的数据类型构成的复杂对象,支持子类、超类和继承的概念,因而能对现实世界的实体进行自然而直接的模拟,可表示诸如某个对象由“哪些对象组成”,有“什么性质”,处在“什么状态”,具有丰富的语义信息,这是传统数据库所不能比拟的。

③提供通用的规则系统。规则在数据库管理系统(DataBase Management System, DBMS)及其应用中是十分重要的,在传统的关系数据库管理系统(Relation DataBase Management System, RDBMS)中用触发器来保证数据库的完整性。触发器可以看成规则的一种形式。OODBS 支持的规则系统将更加通用,更加灵活。例如,规则中的事件和动作可以是任何的 SQL 语句,可以使用用户自定义的函数,规则还能够被继承。这就大大增强了 OODBS 功能,使之具有主动数据库的特性。

3. 网络数据库系统

随着客户机/服务器结构的出现,使得人们可以最有效地利用计算机资源。在客户机/服