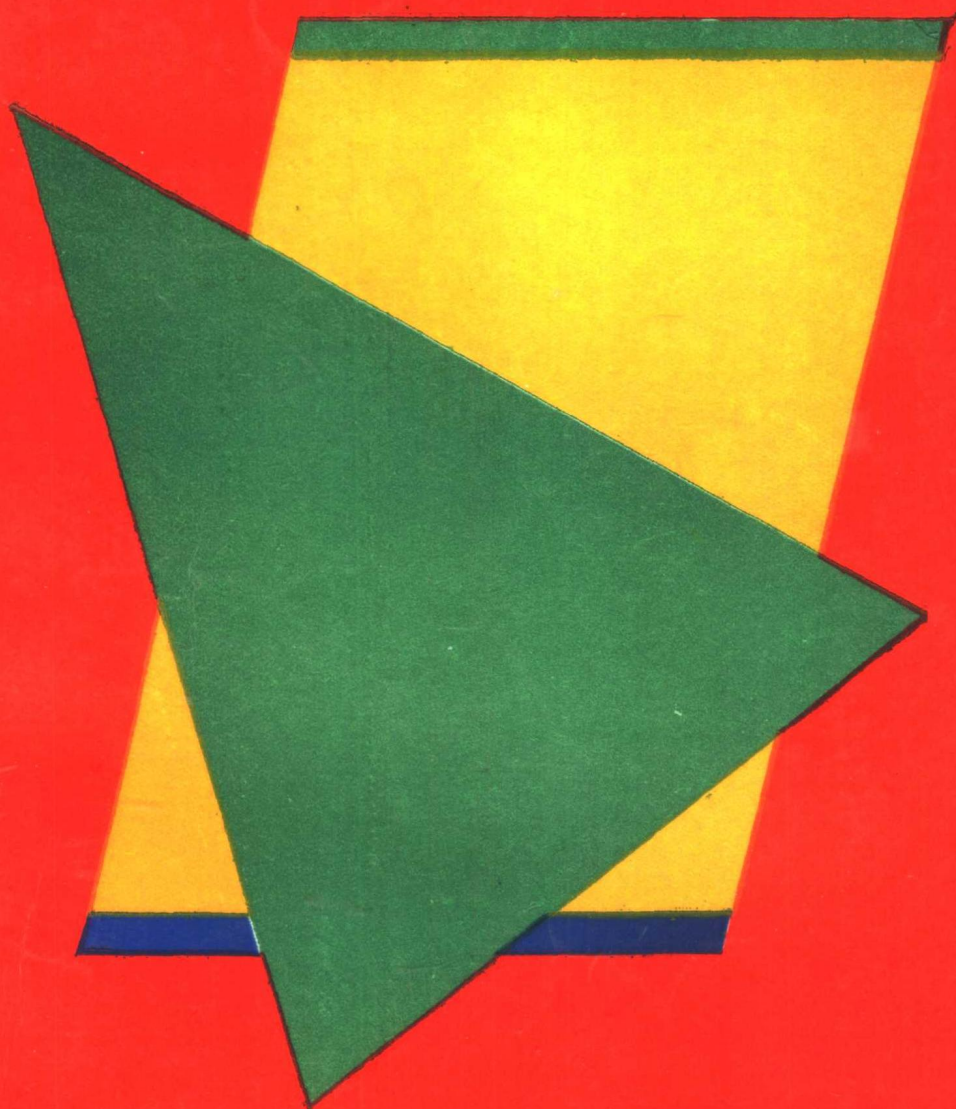


计算机及软件技术丛书

微型计算机编译型关系数据库系统 Clipper 应用程序设计

潘旭燕 潘金贵 萧柔 等编著



南京大学出版社

(苏) 新登字第 011 号

内 容 简 介

Clipper 是与 dBASE III plus、dBASE IV 以及 Foxbase 兼容的微机编译型关系数据库系统。

本书系统地介绍了最新版本的 Clipper 及其程序设计技术。

全书分为四个部分共十四章五个附录。主要内容包括：数据库系统与 Clipper 的基本概念；Clipper 与 DOS 的关系，Clipper 结构化程序设计技术；Clipper 的编译和连接技术；应用程序的调试；Clipper 的错误处理机制；Clipper 的语句、标准命令集和函数详解；建立良好的用户界面的方法与技术；Clipper 工具的使用法；如何在网络环境下使用 Clipper；Clipper 的扩展系统。几个附录则分别介绍了一些对读者有用的附加信息。

本书叙述通俗易懂，内容循序渐进，示例丰富，便于自学，强调 Clipper 程序设计的特点，注重方法与技巧的训练，有助于读者培养良好的程序设计风格。

本书既可作为高等院校计算机专业的本科生和研究生学习 Clipper、dBASE 和 Foxbase 语言的教材或教学参考书，亦可供有关专业的学生以及从事计算机系统软件及应用软件开发工作的工程技术人员作为 Clipper 软件包的使用手册。

欲购买与本书配套的例子盘和最新版本的原版 Clipper 系统盘以及汉化模块的读者可与编者联系获得有关信息。

计算机及软件技术丛书
微型计算机编译型关系数据库系统
Clipper 应用程序设计
潘旭燕、潘金贵、萧柔等 编著

*

南京大学出版社出版

(南京大学校内，邮编 210008)

江苏省新华书店发行 江苏南京通达印刷厂印刷

*

开本：787×1092 1/16 印张：30.5 字数：758 千字

1994 年 3 月第 1 版 1994 年 3 月第 1 次印刷

印数 1~6 000

ISBN 7-305-02447-3 / TP·87

定价 22.40 元

《计算机及软件技术丛书》编委会

学术顾问 孙钟秀 张福炎 郑国梁

主 编 谢 立

副 主 编 时惠荣 潘金贵 丁 益 赵沁平

编 委 (按姓氏笔画为序)

丁 益	丁嘉种	王永成	孙志挥
时惠荣	陈 禹	陈道蓄	赵沁平
杨静宇	钱士钧	钱培德	徐宝文
顾其兵	谢 立	潘金贵	

出版者的话

我国社会主义经济建设的蓬勃发展,极大地推动着社会信息化的进程,也促进了信息产业的发展。现在,计算机的应用已渗透到社会和生活的各个领域。作为社会信息化基础的计算机及软件技术,正为越来越多的人掌握和应用,计算机及软件技术也因此而不断更新、发展。

掌握计算机技术,是现代人特别是跨世纪的中青年人在当今激烈的社会竞争中制胜的基础,也是未来信息化社会对每个人的要求。然而,在我国,计算机基础教育尚欠普及,计算机特别是微型计算机及软件技术的应用和开发也还处在一个较低的层次。许多非专业人员希望能使用计算机,但面对纷繁的专业知识,众多的技术资料,视学习计算机的使用为畏途,专业人员面对软件技术的快速更新,目不暇接。为了让更多的人熟悉计算机技术,利用计算机服务于自己的管理、科研、教学工作,使我国的计算机及软件技术的应用和开发紧随国际潮流,普及和提高我国计算机应用和开发的水平,我们为此组织编写并陆续出版《计算机及软件技术丛书》。

本《丛书》将以应用为基础,兼顾普及与提高。组织科研、教学和应用开发第一线的专家、学者,结合国外计算机及软件技术的最新发展和趋向与国内的应用现状和方向,为初学者提供系统的入门读物,为专业人员介绍适合国情的最新实用技术,既有理论性、学术性强的专著、专论,也有普及性、实用性的教材、手册,以满足多层次读者的需要。

本《丛书》的编写将立足于现实,着眼于未来,力争反映国内外计算机及软件技术的最新动态和发展趋向,引导和帮助读者学习、吸收、掌握计算机的新理论、新技术和新成果。

我们将根据读者需要,不断充实、完善本《丛书》内容,同时诚恳欢迎读者对本《丛书》提出建议、批评,也热忱欢迎向本《丛书》赐稿。

南京大学出版社

《计算机及软件技术丛书》编委会

前言

一、Clipper 系统的特点

随着社会信息化发展的进程,越来越多的计算机走入人们的事务处理和信息管理工作中。就像生活中缺不了水一样,人们的计算机事务管理工作也离不开数据库管理系统。那么,当今众多数据库管理系统中的佼佼者是哪一个呢?

众所周知,由于 dBASE III 数据库管理系统的实用性强,易学易懂,曾被誉为“大众数据库”,广泛应用于各行各业的事务管理中,成为在我国流行最广、应用面最大的微机数据库管理系统。但 dBASE III 也有其固有的弱点,如计算能力弱,处理速度较慢,没有图形功能,没有数组和自定义函数等。这些弱点不但限制了 dBASE III 在更多应用领域的发展,而且导致了用 dBASE III 开发的应用项目只停留在单项程序或比较低的水平上,难以达到综合信息管理的更高要求。近年来,不断有新的与 dBASE III 兼容的数据库推出,如: dBASE III PLUS、FoxBASE、FoxPro、Clipper 等等,其中最成功、性能最优良的当数美国的 Nantucket 公司推出的最新编译型数据库管理系统 Clipper,今天它以其与 dBASE III 和 IV 以及 FoxBASE 完全兼容、功能强、速度快、保密性好等特点而在微机关系数据库管理系统中独领风骚。

dBASE III Plus、FoxBASE 程序像简化版的 BASIC 一样,是解释执行的。也就是说,对程序中的每一行代码,都要对它进行求值,并判断是否有语法错误,没有错误则转换为机器代码,再送到处理器中去执行。这样,如果一个程序循环回到其程序结构的开始,尽管这一段代码在几秒钟前刚被检查过是正确,还需再解释执行一遍。编译型的关系数据库 Clipper 就克服了这一缺点。

所谓编译就是运用编译程序将源程序代码转换为机器语言的过程。编译时,编译程序先检查一遍应用程序(源代码)中每一行代码有无错误。如果没有错误,则将源代码转换为机器语言。机器语言是计算机的微处理器能直接执行的二进制代码。

通常,一个 dBASE 源程序中的命令是以 ASCII 码的形式写到磁盘上的。ASCII 码用数值代替人所使用的字母和标点符号来作为信息符号。编译程序把这些 ASCII 字符转换成处理器能识别的指令。

每一种微处理器都有一个指令集合,用汇编语言命令来表示。编译程序将 ASCII 文件翻译成机器语言后,所得的结果文件就称为目标代码。

目标代码要和其他几段机器语言连接,这几段机器语言包含了使程序员的目标文件完整所需要的指令和例程。所得结果被写到一个以 .EXE 为扩展名的文件中,该扩展名表示该文件是一个可执行程序。

编译过的程序的运行速度可以比一个解释的程序快上十倍或更多。但不能改变编译过的程序中的代码;若要改变某些程序行,则必须修改源代码。

而在解释执行的系统中运行一个程序，需要这种语言解释程序的一个副本。例如，为运行一个用 BASIC 语言编的程序，必须先调用 BASIC 解释程序，并把 BASIC 程序源代码装入内存，然后告诉 BASIC 去运行。而编译生成的执行文件，是独立的，它可以在操作系统提示符下键入文件名并按回车来运行它。不需运行解释程序或编译程序，就能运行程序。可是通过编译只生成.OBJ 文件，要生成.EXE 可执行文件，还要连结各.OBJ 文件。

连结是由连接程序来完成的，连接程序的功能是将各.OBJ 文件组合起来，形成一个可执行的.EXE 文件。

Clipper 是一种开发者的工具，它没有像在 dBASE III Plus、dBASE IV 和 Foxbase 中那样使用点提示符或控制中心来为用户提供一个交互式的界面，而是侧重于编制数据库应用程序的工作，这些应用程序是由程序员从最底层开始设计，编译并连接成一个可执行程序，然后提交给用户。用户不必知道有关 dBASE 的知识，他们所需知道的就是如何使用程序。这样可以大大地提高程序开发效率。

Clipper 是 dBASE 系列的一个编译版本，它的命令和函数是 dBASE III Plus 的超集。Clipper 是第一个提供用户定义函数的 dBASE 语言产品；从 5.0 版本起，Clipper 又第一个提供了用户自定义命令，可供 Clipper 程序设计人员根据需要来改造 Clipper 语言。

另外，Clipper 的扩展系统使程序员和用户可以很容易地访问用 C 语言和汇编语言编制的例程和函数。只要程序员发现 Clipper 中缺省什么，就可以很方便地自己补上。

最新版本的 Clipper 又一次在技术上有了新的突破。它推出了嵌入式的处理程序，使得用户可以建立自己定义的命令条件编译，以完全控制 Clipper 环境；它还推出了一个新的连接程序，并动态地分配内存，这样多个非常大的程序也能在有限的内存上运行。这个新的连接程序的功能之一是它能够建立预先连接好的库 (Pre-Linked Library)。PLL 是已连接好的程序代码文件，如果需要可以被装入内存。PLL 可以大到写满它们所在的磁盘，在运行时只有把需要的部分代码装入内存。这意味着用户可以在一台只有 512K RAM 的机器上运行一个 900K 的程序。

除了以上这些特点，Clipper 5.X 版还具有使用非 dBASE III 兼容数据文件的能力；还包括了一个高效、用户可存取的编译预处理程序，一个新的连接程序和一个新的调试程序。使用 Clipper 可以把编译好的应用程序提交给用户而无需为每一个用户都准备一个 dBASE 副本。

二、Clipper 系统运行环境

从技术上说，使用 Clipper 只需要一台带 256K 内存和单色显示卡的 PC 兼容机和 DOS 2.0 或更高版本的支持。为合理使用 Clipper，最好有一台 AT 型的 80286，80386 机器或兼容机。如果打算编写网络应用程序，则需要 DOS 3.1 以上版本支持。Clipper 需要 DOS V3.3 版本才能发挥全部作用。

经过编译后的 Clipper 应用程序可在满足下列条件的计算机上运行：

- PC-DOS 或 MS-DOS 2.0 版以上的单用户系统。
- PC-DOS 或 MS-DOS 3.1 版以上的局域网络多用户系统。
- 256KB 以上 RAM。

· IBM PC / XT或AT及其全兼容的计算机,或是带ANSI终端的任何计算机。

三、本书的组织

自 1987 年夏季推出 Clipper 的早期版本以后,Clipper 系统已经经历了重大发展,版本多次更新,深受广大 dBASE 程序员和用户的欢迎。目前,国内外已经有众多的数据库程序设计人员开始使用 Clipper 软件。

为了使我国广大的计算机工作人员能够更好地利用 Clipper 进行程序设计工作,在南京大学出版社的支持下,我们根据从事用 dBASE、FoxBASE 和 Clipper 语言程序设计的经验,以 Clipper 的最新版本为蓝本,参阅了国内外的原版软件和有关资料,编写了这本《微型计算机编译型关系数据库系统 Clipper 应用程序设计》。它全面、系统地介绍了 Clipper 的基础知识和高级程序设计技术,是一本引导读者进行 Clipper 程序设计的较好的参考书。

本书由十四章和五个附录组成。

第一章介绍数据库系统的基本概念;第二章概要介绍 Clipper 程序设计基础;第三章介绍 Clipper 结构程序设计;第四章介绍 Clipper 应用程序的编译与连接;第五章介绍如何调试 Clipper 应用程序;第六章介绍 Clipper 的错误处理;第七章介绍用户接口程序设计;第八章介绍 Clipper 的有关工具;第九章介绍了如何在 DOS 环境下更好地使用 Clipper;第十章则介绍 Clipper 在网络环境下的使用;第十一章介绍 Clipper 与其他语言的接口;第十二、第十三、第十四分别详细地介绍了 Clipper 的语句、标准命令集和函数。

附录 A 列出了 INKEY()函数的返回值;附录 B 列出了 Clipper 的保留字;附录 C 说明了 dBASE III Plus 与 Clipper 的区别;附录 D 详细列出了 Clipper 的所有错误信息;附录 E 给出了 Clipper 语句、标准命令与函数索引,以方便读者查找这些语句、命令和函数。

参加本书编著工作的主要人员有潘旭燕、潘金贵、萧柔等;朱训衷、李灵君、王勇、张岗、吴卫华、顾铁成、于宇新、冀惠刚、忻超、萧华、金子方等,也作了不可缺少的工作。潘金贵和萧柔同志对全书进行了仔细的修改和统编。

本书承蒙国内较早从事数据库系统教学和研究的南京大学计算机科学与技术系徐洁盘教授主审,顾铁成同志校阅了全书,在此谨向他们深表谢意。

由于水平、时间所限,不妥之处在所难免,欢迎广大读者批评指正。

编著者

1993 年 6 月于南京

目 录

第一部分 Clipper 程序设计基础

第一章 数据库系统的基本概念	3
1.1 数据库技术的发展过程	3
1.2 数据库系统的特点	4
1.3 数 据 模 型	6
1.4 数据库管理系统的功能与构成	10
1.5 Clipper 和关系运算	11
1.6 数据库设计	12
1.6.1 需求分析	13
1.6.2 概念结构设计	14
1.6.3 逻辑结构设计	16
1.6.4 物理结构设计	19
1.7 数据库保护	20
1.7.1 安全性	20
1.7.2 完整性	21
1.7.3 并发控制	21
1.7.4 数据库恢复	22
第二章 Clipper 程序设计基础	23
2.1 Clipper 的系统参数	23
2.1.1 Clipper 数据库文件的结构	24
2.1.2 数据库字段	25
2.1.3 字段名	25
2.1.4 字段的类型	26
2.2 内存变量	26
2.2.1 数组内存变量	27
2.2.2 内存变量和数组的作用域	27
2.3 运算符	28
2.3.1 数学运算符	28
2.3.2 关系运算符	29
2.3.3 逻辑运算符	29
2.3.4 字符串运算符	31
2.3.5 赋值运算符	31

2.3.6	加/减运算符	32
2.3.7	特殊运算符	32
2.4	表达式	32
2.5	宏代换	33
2.6	光标键的使用	33
2.6.1	全屏幕操作的光标移动键	33
2.6.2	全屏幕编辑键	34
2.6.3	全屏幕退出键	34
2.6.4	全屏幕模式键	34
2.7	安装 Clipper	34
第三章	Clipper 结构化程序设计	36
3.1	结构化程序设计方法	36
3.1.1	程序设计的步骤	36
3.1.2	结构化程序设计方法	36
3.1.3	结构化程序设计的原则	37
3.1.4	算法的描述工具	38
3.2	程序文件的建立、修改和运行	40
3.2.1	程序文件的建立与修改	40
3.2.2	程序文件的运行	41
3.3	程序文件中的交互式命令与辅助命令	42
3.4	顺序程序设计	45
3.5	分支程序设计	46
3.5.1	简单选择语句	46
3.5.2	多分支选择语句 DO CASE-ENDCASE	49
3.6	循环程序设计	52
3.6.1	DO WHILE-ENDDO 语句的格式	52
3.6.2	循环结构的基本类型	52
3.6.3	多重循环	55
3.6.4	使用循环语句应注意的问题	60
3.6.5	FOR...NEXT 循环	61
3.7	过程与过程调用	62
3.7.1	过程及过程调用的基本概念	62
3.7.2	过程文件	63
3.7.3	过程文件的打开和关闭	64
3.7.4	过程调用与内存变量的属性	65
3.7.5	带参数的过程调用	67
3.8	用户自定义函数	70
3.9	错误捕获与键盘侦测	73
3.9.1	ON ERROR	73

3.9.2	ON KEY	76
3.9.3	ON ESCAPE	78
第四章	应用程序的编译与连接	79
4.1	由源代码生成.EXE 可执行文件	79
4.2	应用程序的编译和连接	80
4.2.1	控制编译程序	81
4.2.2	指定编译程序开关项设置	84
4.2.3	利用.CLP 文件进行编译	84
4.2.4	建立.CLP 文件	84
4.2.5	编译结束代码	85
4.3	安 装 RTlink	85
4.3.1	RTLINK.CFG	86
4.3.2	RTLINKCMD 变量	86
4.3.3	RTlink 命令行	86
4.3.4	从脚本文件中设置选项	86
4.4	使 用 RTLINK	86
4.4.1	RTLINK 和 DOS 的 ERRORLEVEL	87
4.4.2	把连接信息存入文件	87
4.4.3	操作方式	87
4.4.4	RTLink 关键字	88
4.4.5	增量连接 (Incremental Linking)	90
4.4.6	预连接库 (PLL)	91
4.4.7	确定预连接库的内容	92
4.4.8	建立用于开发的 PLL	92
4.4.9	建立提交用户的 PLL	93
4.4.10	建立简单的提交用户的 PLL	94
4.5	使用 RMAKE 工具	95
4.5.1	RMAKE 处理	95
4.5.2	双重目的的 MAKE 文件范例	100
第五章	应用程序的调试	101
5.1	在 DOS 提示符下调用调试工具 Debugger	101
5.2	调试工具 Debugger 的窗口	102
5.3	功能键	103
5.4	调试工具 Debugger 的菜单	104
5.4.1	选项 File 子菜单	105
5.4.2	选项 Locate 子菜单	105
5.4.3	选项 View 子菜单	106
5.4.4	选项 Run 子菜单	107
5.4.5	选项 Wacth 子菜单	108

5.4.6 选项 Monitor 子菜单	109
5.4.7 选项 CallStack 子菜单	109
5.5 监视点、跟踪点、通过点和断点	110
5.6 调试器命令	111
5.7 调试处理过程	115
5.8 浏览其他文件	116
5.9 使用数组	116
5.10 处理运行时发生的错误	117
第六章 Clipper 的错误处理	118
6.1 错误的捕获	118
6.2 使用 BEGIN SEQUENCE 结构	120
6.3 Clipper 的错误对象	121
6.3.1 错误对象	121
6.3.2 ERRORBLOCK()函数	122
6.4 面向对象的错误处理程序	122
6.5 Clipper 的错误信息	125

第二部分 Clipper 高级程序设计技术

第七章 用户接口程序设计	129
7.1 简单的菜单	129
7.2 BOX	133
7.3 按键的处理	138
7.4 光标的处理	142
7.5 填充键盘缓冲区	143
7.6 SAVE SCREEN / RESTORE SCREEN	150
7.7 屏幕和 MEM 文件	154
7.8 GET 的处理	155
7.8.1 UPDATED	155
7.8.2 VALID	157
7.8.3 用 VALID 来处理	160
7.8.4 诀窍	161
7.8.5 二维的 GET	163
7.9 垂直滚动	167
7.9.1 诀窍	167
7.9.2 浏览数据	167
7.10 对话框	173
7.10.1 浏览简单的数据文件	176
7.10.2 浏览复杂的数据文件	180

第八章 Clipper 的工具	184
8.1 Clipper 的文件	184
8.2 Clipper 的实用程序	185
8.3 DBU 实用程序的使用	185
8.4 使用 RL 实用程序	186
8.4.1 生成报表	187
8.4.2 建立标签文件	188
8.5 PE 实用程序的使用	189
第九章 更好地使用 Clipper	190
9.1 Clipper 的内存管理	190
9.1.1 非常规内存管理	191
9.1.2 虚拟存储管理	192
9.2 DOS 环境的控制	192
9.2.1 用启动文件控制 DOS	193
9.2.2 用变量控制 DOS	193
9.2.3 SET CLIPPER 变量	193
9.2.4 把 DOS 变量传递到 Clipper 应用程序	195
9.2.5 安装 Clipper	196
9.2.6 提供 DOS 服务	196
9.3 DOS 开发环境	197
第十章 在网络环境下使用 Clipper	198
10.1 多用户程序设计	198
10.2 Clipper 的网络兼容性	199
10.3 Clipper 与网络有关的命令	200
10.4 开发网络应用程序	200
10.4.1 设计程序时要考虑到网络的多用户性	201
10.4.2 选择处理方式	201
10.4.3 上锁失败时的策略	201
10.5 在局域网上打开文件	201
10.5.1 用 USE__UDF()函数打开一个文件	202
10.5.2 用 APND()函数增加一个新记录	204
10.5.3 用 RLOK()函数给当前记录加锁	205
10.5.4 用 FLOK()给当前文件加锁	207
10.6 记录和文件上锁的实现	208
10.7 非数据和索引文件	208
10.8 有关文件的操作命令和函数	209
第十一章 Clipper 与其他语言的接口	210
11.1 Clipper 的扩展系统是什么	210
11.2 为什么在编程过程中要使用 C 或汇编语言	210

11.3 Clipper 与 C 的接口	210
11.3.1 以值的形式传递参数	211
11.3.2 返回参数给 Clipper	211
11.3.3 以指针的方式传递参数	212
11.3.4 内存分配	212
11.3.5 Extend.h 头文件	213
11.3.6 C 函数模板	213
11.3.7 C 函数样本程序	214
11.3.8 与 Clipper 一起编译 C	216
11.3.9 有关 C 的其他版本的一些说明	217
11.4 C 和汇编语言的函数	217
11.4.1 样本程序框架	218
11.4.2 有关 MASM 函数的例子	218
11.4.3 编译、汇编及连接	224
11.4.4 调试汇编程序	224

第三部分 Clipper 语句、命令和标准函数参考

第十二章 Clipper 语句	227
第十三章 Clipper 标准命令集	248
第十四章 Clipper 标准函数	329

第四部分 附 录

附录 A INKEY()函数返回值	441
附录 B 保留字	442
附录 C dBASE III 和 Clipper 的区别	445
附录 D Clipper 错误信息	453
附录 E Clipper 语句、标准命令与函数索引	469

第一部分

Clipper 程序设计基础

第一章 数据库系统的基本概念

本章主要阐述有关数据库的基本概念,如数据库的特点、数据库管理系统 (DBMS)、数据模型、数据库系统的种类等;还要阐述数据库的一些理论知识,如关系数据库的理论基础、数据库保护机制以及数据库设计所用的实体-联系方法 (E-R 方法);此外,讨论了数据库的发展,并在论述过程中简单介绍目前最新的几种大型数据库系统,如 ORACLE, Informix, Sybase 等,这些内容是学习编译型数据库管理系统 Clipper 的理论基础。

1.1 数据库技术的发展过程

数据库系统是现代计算机系统的一个重要组成部分,是人们高效地进行数据存储、共享和处理数据的工具。现代计算机已不再仅仅应用于科学计算,而是更广泛地应用在各种管理工作中,进行大量的数据和知识处理。在管理活动中要涉及大量数据的存储、共享、加工和传输,因此,要使管理工作现代化,并用计算机辅助决策,就首先要有一种工具来管理大量的信息,这种客观要求导致数据库这门技术产生并迅速发展。

早期进行大量的数据存储与管理都是使用文件系统。但随着联机实时处理的增多、数据应用的复杂化,文件系统表现出了许多缺点:

- 数据冗余度大,因而空间浪费、存取时间长、数据间容易出现不一致和不相容。
- 数据和程序间不具有独立性,两者相互依赖,因而系统不易扩充、不易维护,数据不易共享。

数据库系统就是在文件系统的基础上发展起来的,它克服了文件系统的缺点,其发展也日趋完善,经历了如下几个发展时期。

1. 摇篮时期: 60 年代

60 年代中期,数据库技术开始萌芽,数据库概念开始形成,较有影响的工作包括:

- 1968 年,网状数据库系统 TOTAL 等开始出现。
- 1969 年,McGee 等人开发的层次数据库系统、IBM 公司的 IMS 发表。

2. 发展时期: 70 年代

在这一时期,数据库技术日益发展,并有了坚实的理论基础。

(1) 美国数据系统语言协商会下属的数据库任务组 DBTG (DataBase Task Group) 对数据库方法进行研究后提出了 DBTG 报告。从而确定了数据库的许多方法和技术。DBTG 是基于网状结构的,是数据库网状模型的基础和典型代表。

(2) 1970 年 IBM 公司 San Jose 研究所的研究员 E.F. Codd 发表了题为“大型共享数据库数据的关系模型”论文,提出了数据库的关系模型,开创了数据库关系方法和关系数据理论的研究,为数据库技术奠定了理论基石。由于 E.F.Codd 的杰出贡献,他于 1981 年获得了 ACM 图灵奖。

3. 成熟与集成时期: 80 年代至今

这一时期, 数据库技术迅速发展, 大量商品化的数据库系统广泛应用于各种领域、各种硬件环境, 数据库理论更加充实, 并且伴随着计算机技术与通讯技术的发展, 数据库系统产生了许多集成产品。这一阶段的成果表现在:

(1) 丰富的关系型数据库产品。从大型机到微机, 从 UNIX 操作系统到 DOS 操作系统, 推出了许多出色而成熟的数据库系统, 比如: ORACLE, INFORMIX, SYBASE, UNIFY/ACCELL, PROGRESS, INGRES, RDB, PARADOX, dBASE, FoxBASE, FoxPro, 编译型 dBASE, 编译型 Clipper 等等。

(2) 由于网络通信技术的发展, 在集中式数据库系统成熟技术的基础上产生和发展了分布式数据库系统 (Distributed Database System), 网络节点上的用户可以存取异型机上的任何数据而不用知道该数据存储在哪一物理设备上。因而, 大型的数据库系统都推出了相应的网络和分布式版本。比如 INGRES 的分布式版本 INGRES/STAR, INFORMIX 和 ORACLE 的 SQL*Net, SQL*STAR 等, 从而实现了新型的顾客/服务器模式 (Client/Server)。

(3) 为了提高数据库应用系统的开发效率、提高软件的生产率, 许多软件公司基于数据库系统生产了大量的 CASE (Computer Assisting Software Engineering) 工具、第四代软件开发环境和第四代语言 (Fourth Generation Language, 又称面向问题语言)。比如: INFORMIX 的 4GL 及 OA 软件, ORACLE 的 CASE 产品系列, FoxBASE 的 FoxView, FoxCode, FoxDoc, FoxGraph 等。

(4) 随着数据库应用领域的不断扩大, 数据库系统已能进行多媒体 (Multimedia) 处理, 不仅能处理常规的数据, 而且还能处理图形、图像、声音等, 如 INFORMIX 的 Online 等。

1.2 数据库系统的特点

数据库可以视为“管理大量的、持久的、可靠的、共享的数据之工具”。从这一简单定义中可以看出, 数据库系统是一种管理数据的工具, 它的管理对象具有以下特征:

(1) “大量”, 之所以称为数据库就是因为是大量数据的集合, 这表明数据量大, 不能放在通常的主存中, 需要大容量的外存作支持。

(2) “持久”, 意思是指数据必须长久地保留, 这表明数据不是简单地为一特定应用准备的, 不是像科学计算那样处理完成数据就随之消失。

(3) “可靠”, 是指一旦系统发生可能的软、硬件故障, 可以恢复数据库。

(4) “共享”, 是指若干用户能按照一定有序的方式存取数据库中的数据, 避免同步存取可能造成的错误。

与文件系统相比, 数据库系统有如下显著的特点:

1. 数据结构化

在文件系统中, 从整体上讲, 数据是无结构的, 即文件的记录型之间没有联系。文件系统只关心记录内数据项之间的联系, 它也无法实现文件之间的联系。

数据库系统则不仅要考虑数据项之间的联系, 更要考虑记录型之间的联系 (是一对一、