

DATA BASE IV 程序设计 及几种小型数据库使用简介



中国科学院希望高级电脑技术公司

DATA BASE IV 程序设计 及几种小型数据库使用简介

中国科学院希望高级电脑技术公司

db4 db4 db4 db4 db4 db4 db4 db4 db4 db4 db4 db4 db4 db4 db4 db4 db4 db4

前 言

dBASE II 和 **dBASE III** 的应用, 对于微机的用户单位, 已近乎家喻户晓。但其汉字的输入, 多采用拼音法等。其原因是 **dBASE III** 的程序结构较大, "王码" 汉字输入软件, 难于注入内存。目前社会上有相当数量的微机, 其内存仅 512K RAM, 或单软驱, 或双软驱。为了在小配置的系统中, 能方便地使用 "王码" 五笔字型输入法来建立小型中文关系式数据库, 我们推荐采用 **DATA BASE IV**。

DATA BASE IV 简称 **db4**。是 1985 年由 Sichuan 计算机服务公司推出的。是续 **dBASE II** 和 **dBASE III** 之后引进的一种新型个人计算机用的关系型数据库管理系统, 能直接运行汉字。可在 **IBM PC/XT**; **IBM PC/AT**; **IBM 5550** 及国产 05 系统的各种兼容微机上使用。可由 "王码" 的五笔字型法来输入各种汉字数据项。能在汉字磁盘操作系统 **CCDOS 2.0** 以上的版本上运行。它容易自学, 灵活方便, 功能很强, 近乎 **dBASE III** 的功能, 适于推广应用。用于建立完整的数据库系统, 进行各种数据处理都很得心应手, 是一种较好的应用软件开发工具。适用于商业, 财贸, 企业管理和小型事务处理上。

db4 突出的特点如下:

- 1, 能同时打开十个中文数据库文件, 对其连接与分解的能力强, 可共享十个文件中的数据, 相互调用。
- 2, 能与高级语言方便地结合使用。
- 3, 命令条数较少, 仅 44 条。比 **dBASE III** 几乎少了一半。故简明易学易掌握。其一条命令有多种功能, 可予选用。
- 4, 能方便地调用外部文件。在 **db4** 状态下可直接使用操作系统命令。
- 5, 与 **dBASE III** 本比, 具有较多的初等函数运算功能。
- 6, 程序结构较小, **db4.exe** 仅有 116KB, 故可将 "王码" 软件与 **db4** 同时装入内存而采用五笔字型法输入汉字。
- 7, 可在直接命令方式和程序方式下运行, 且编程十分简单。

我们希望大家不防用一用 **db4**, 您会感到省事, 而且高效。所以我们编了这本册子, 以协助大家使用 **db4**。

参加本书编写工作的有: 上海交大微机研究所姚萝姑。陕西机械学院刘凯, 张海燕。新疆轻工设计研究院计算机室任一民。新疆气象科研所计算机室刘策等同志。

参加本书分章校审工作的有: 孙振铎, 刘继明, 羊勇生, 张瑞兵, 杨林波斯坦, 葛奇等同志。

本书中的错误和不妥之处, 恳请读者指正。

任 干 生

一九九零年二月二十日。

目 录

第一章 DATA BASE IV 概述	1
1-1 db4 的特点	1
1-2 db4 的环境	4
1-3 db4 的结构和性能	4
1-4 db4 的操作方式	5
1-5 db4 的语句	5
1-6 db4 的文件	9
1-7 db4 的数据类型	11
1-8 db4 的变量	11
1-9 db4 的表达式	14
1-10 对记录量的限制	15
第二章 db4 的函数	18
第一节 数值型函数	18
2-1-1 自然指数函数(exp)	18
2-1-2 取整数函数(fix)	21
2-1-3 取整数函数(int)	21
2-1-4 自然对数函数(log)	22
2-1-5 平方根函数(sqr)	26
2-1-6 求绝对值函数(abs)	29
2-1-7 求反正切函数(atan)	30
2-1-8 求余弦函数(cos)	30
2-1-9 求正弦函数(sin)	31
2-1-10 求正切函数(tan)	35
2-1-11 判符号位函数(sgn)	40
第二节 字符串型数据运算函数	41
2-2-1 字符串中小写变大写的函数(cap\$)	41
2-2-2 宏代换函数(&)	41
2-2-3 建立空格变量函数(space\$)	52
2-2-4 子串查找函数(instr)	53
2-2-5 移去字符串尾部空格的函数(trim\$)	53
2-2-6 求字符串长度的函数(len)	53
2-2-7 取字符串左端字符的函数(left\$)	53
2-2-8 取字符串右端字符的函数(right\$)	54
2-2-9 取字符串中间字符的函数(mid\$)	54
2-2-10 取字符串中首字符个数的函数(string\$)	57
第三节 日期函数	58
2-3-1 显示当前日期的函数(date\$)	58

2-3-2 显示当前时间的函数(time\$).....	61
第四节 转换型函数.....	62
2-4-1 求字符 ASCII 码的函数(asc).....	62
2-4-2 由 ASCII 码求字符的函数(chr\$).....	62
2-4-3 数值型转换成字符型的函数(str\$).....	63
2-4-4 字符型转换成数值型的函数(val).....	64
2-4-5 十进制数转换成十六进制数的函数(hex\$).....	64
第五节 状态检验函数.....	65
2-5-1 文件结束函数(eof).....	65
2-5-2 删除文件的函数(del).....	66
2-5-3 检查文件的函数(file).....	67
2-5-4 查当前记录的函数(loc).....	67
2-5-5 求数据文件中记录总数的函数(lof).....	67
2-5-6 find 命令中对键值的记录找到与否的函数(match).....	67
第六节 查内存的函数.....	68
2-6-1 求内存中特定地址的内容(peek).....	68
第三章 db4 数据库的操作命令.....	69
第一节 进入与退出 db4 及建立数据库结构.....	69
3-1-1 进入和退出 db4 (sys).....	69
3-1-2 建立 db4 数据结构(define).....	70
第二节 数据库中记录的录入.....	73
3-2-1 录入数据时应注意的事项.....	73
3-2-2 数据文件的打开(open)和关闭(close).....	74
3-2-3 向已打开的数据文件追加记录(append).....	75
3-2-4 修改数据文件中的数据.....	86
3-2-5 向数据文件中写入数据(write).....	90
3-2-6 数据文件记录置空(blank).....	92
第三节 数据库的组织与记录的查寻.....	93
3-3-1 记录指针的定位(read).....	93
3-3-2 数据库的分类与排序(sort).....	95
3-3-3 建立观察文件(view).....	99
3-3-4 查找数据库中的记录(find).....	102
第四节 数据库中数据记录的删除(delete).....	104
3-4-1 加删除标记(delete).....	104
3-4-2 恢复记录(recall).....	106
3-4-3 删除记录(pack).....	106
第五节 数据文件的显示与列表.....	107
3-5-1 显示磁盘文件目录(files).....	107
3-5-2 屏幕显示命令(display).....	108
3-5-3 屏幕编辑命令(screen).....	109

第六节 数据库的复制和结构的修改(related)	113
3-6-1 复制整个数据库文件	113
3-6-2 复制数据库文件的部分数据	113
3-6-3 复制数据库文件的结构	115
3-6-4 在屏幕上修改数据文件结构(modify)	116
3-6-5 从一个数据文件向另一个数据文件追加数据	117
3-6-6 将数据文件转换成 ASCII 码文件(related-text)	118
3-6-7 改字段名编辑	122
3-6-8 不改字段名编辑	124
第七节 数据库中数值型数据的计算	127
3-7-1 求和命令(sum)	127
3-7-2 小计命令(total)	128
3-7-3 记录数计数命令(count)	129
第八节 db4 与其它语言的结合使用	130
3-8-1 db4 与 dBASE II 文件的转换	131
3-8-2 db4 与 BASIC 语言数据的结合使用	132
3-8-3 db4 与 forTRAN 语言数据的结合使用	135
第四章 建立数据报表(report)	137
第一节 建立数据报表文件	137
第二节 数据报表的输出	141
第五章 关系型数据文件间的操作	142
第一节 概述	142
第二节 在数据文件间建立关联(related-join)	142
第三节 更新数据库的操作(related-att)	147
第四节 移动两个数据文件中的联接	148
第六章 辅助操作	151
第一节 变量的使用	151
6-1-1 定义内存变量(=)	151
6-1-2 显示内存变量(?)	151
6-1-3 清除内存变量(clear)	152
第二节 文件操作命令	154
6-2-1 列文件目录	154
6-2-2 清除文件(kill)	155
6-2-3 文件改名(rename)	155
6-2-4 关闭文件(close)	156
第三节 其它命令	156
6-3-1 //命令	156
6-3-2 屏幕上显示信息(beginwp-endwp)	157
6-3-3 清除命令(cls)	157
6-3-4 设置段地址(defseg)	157

6-3-5 向内存某地址写数据(poke).....	158
第七章 db4 高级语言的结构.....	159
第一节 db4 命令文件概述.....	159
第二节 db4 命令文件的建立与执行.....	159
7-2-1 命令文件的建立.....	159
7-2-2 命令文件的执行 (/).....	160
第三节 db4 命令文件的结构及命令文件的顺序执行.....	164
7-3-1 db4 命令文件的结构.....	164
7-3-2 db4 命令文件的顺序执行.....	165
第四节 db4 命令文件的判断分支.....	166
7-4-1 简单判断语句(if—endif).....	166
7-4-2 选择判断语句(if—else—endif).....	167
7-4-3 选择判断语句的嵌套.....	168
7-4-4 结构式判断执行语句(do case—endcase).....	170
7-4-5 无条件分支转移命令(goto).....	174
第五节 db4 命令文件的重复执行(do—while—enddo).....	174
第六节 db4 命令文件的子程序调用.....	178
7-6-1 调用子程序.....	178
7-6-2 转子程序命令(gosub).....	179
7-6-3 程序结束命令(end).....	180
第七节 db4 系统参数的设置(change).....	180
第八节 db4 命令文件中嵌入高级语言.....	182
第九节 命令文件设计与执行的实例.....	183
第八章 输入与输出格式设计.....	194
第一节 输入语句.....	194
8-1-1 注释命令(!).....	194
8-1-2 终止命令(stop).....	194
8-1-3 暂停命令(get).....	194
8-1-4 键盘输入命令(input).....	194
8-1-5 kin 赋值命令(take).....	195
第二节 屏幕格式控制命令及格式文件.....	208
8-2-1 屏幕坐标.....	208
8-2-2 屏幕格式控制命令(locate).....	208
8-2-3 屏幕格式文件的建立.....	212
第三节 输出语句(print).....	214
第九章 DB 屏.....	216
第一节 DB 屏概述.....	216
第二节 DB 屏操作方法.....	216
9-2-1 启动 DB 屏.....	216
9-2-2 建立屏图.....	217

9-2-3 DB 屏编辑命令	218
9-2-4 变量与标志信息的输入	220
9-2-5 定义命令文件名	221
9-2-6 定义数据文件名	222
9-2-7 DB 屏操作小结	222
9-2-8 DB 屏的错误信息一览表	223
第十章 DBED.exe 编辑文件	225
第一节 DBED 编辑概述	225
第二节 启动 DBED 编辑	225
10-2-1 在 DOS 操作系统下启动	225
10-2-2 在 db4 下启动	225
第三节 DBED 的编辑功能	226
10-3-1 小操作基本规则	226
10-3-2 屏幕滚动 scrolling 命令	227
10-3-3 显示 display 和打印 print 命令	227
10-3-4 复制 copy 和移动 move 命令	227
10-3-5 插入 insert 和删除 delete 命令	228
10-3-6 查找 find 和替换 replace 命令	229
10-3-7 制表 tab 命令	230
10-3-8 转换 transfer 命令	230
10-3-9 退出 exit 和停止 quit 命令	230
第十一章 各种 dBASE 命令简介	232
第一节 dBASE II 命令简介	232
第二节 dBASE III 命令简介	239
第三节 FOXBASE 命令简介	249
第四节 编译 dBASE III 命令简介	256
第五节 dBASE III PLUS 命令简介	265
第六节 dBASE IV 命令简介	279
附录 A db4 命令和语句一览表	304
附录 B db4 的出错信息表	308
附录 C “王码”汉字输入法简介	309
附录 D 行编辑 EDLIN.exe 简要说明	312
结束	313

第一章 DATA BASE IV 概述

计算机数据处理系统,是利用计算机的高速算术和逻辑运算功能,数据传送功能和大容量的存贮功能,对人们搜集和积累的各类数据信息进行加工,保存和管理的过程。我国在六十年代前,多由人工对数据进行处理。随后,由于计算机及其磁盘存贮器的发展,管理软件的文件系统的发展,得使数据以文件的形式保存在磁盘中,以便于对数据进行查询,修改,增删和分类。这时的程序和数据已相对独立。可由文件为单位来共享数据。然而对文件中的记录仍无法访问。进入到七十年代后,发展了数据库管理系统。才得以对数据的管理涉及到文件中的记录,甚至能达到某个字段,从而大大地方便了对数据的管理和使用。

数据库是按某种最优组合方式,将存贮在一起的相互关联的数据的一个集合。数据库系统由文件管理发展而来。文件中的数据只能为某个程序所独用。如 BASIC 语言中的 Data 语句所引导的数据,只能为该应用程序所独用。而数据库中的数据,通过不同的组合形式,可同时为多个应用程序和各个用户所共享。这种组合形式由数据库管理系统所决定,如 db4 数据库管理系统就具备这种功能。

关系型数据库的数据模型是一个关系模型,数据以一张二维表的形式表示出来,即日常事物中的大量表格,如工资表、职工花名册表、财务账页、各式材料清单表、各种科学试验和测量的结果数据表等等。表中各列称为字段,每个字段在各种关系表中有其自己的名字,如工资表中的姓名、基本工资、附加工资、奖金、房费、水电费等等均为字段名。表中的一行为一条记录,它是字段的一种集合。这样构成的数据表框架,叫做文件结构。

对关系型数据库的管理,即数据库管理系统,它是一种应用软件。如 vBASE II、dBASE III、db4 等等都是小型数据库管理系统。目前,对于 dBASE III 的应用,多数微机工作者已相当熟悉了。db4 是继 dBASE II 和 dBASE III 之后引入我国的。两者很相似,但各有其优点,根据用户的实际情况来选用那一种,将会得到最佳的效果。本书中介绍的是一九八五年由 Sichuan 计算机服务公司推出的 1.1 版,其主要性能介绍于下。

1-1 db4 的特点。

Data Base IV, 简称 db4,它是随着数据库技术的应用而发展起来的一种专供个人计算机使用的关系型数据库管理系统。做为一种程序设计的工具,在国外已被广泛地应用。由于它的功能很强,它可以建立一个完整的数据库系统;可做为一种软件开发工具,以缩短程序开发的时间;它可做为范围广泛的通用商业软件基础或事物管理软件基础;可方便地相互共享其它语言程序的数据;通过使用变量,做为专门计算数据的基础;能在简单配置的低挡次微机上运行。它除了具备 dBASE III 的基本功能外,还具有以下特点。

(一) 直接对汉字记录进行处理,由于 db4 命令的组合功能较强,故其程序结构较小。db4.exe 文件只有 116KB。可与软盘“王码”五笔字型同时装入 512KB 的内存中,用“王码”输入法来输入汉字。这一点在 dBASE III 上是不能实现的,dBASE III 必须使用 640K 的

内存配置。由于目前社会上仍有大量的 512K 配置的微机尚没有扩展内存，若采用 dBASE III 来做为汉字数据库管理系统。就只能使用 CCDOS 的拼音输入法、或是首尾输入法等等，无疑是输入速度较慢。而五笔字型输入法已成为我国主要的汉字输入法，普及面很广，要用“王码”五笔字型输入法在 512KB 的微机上建立小型中文关系式数据库，似乎目前只好采用 db4 为宜。

(C) db4 能方便的调用其它外部文件。db4 是由 MS-DOS 或 CCDOS 支持的。在进入 db4 状态后，只要使用“//”命令，就能直接键入并运行操作系统的全部命令包括 DOS 的内部命令和外部命令；包括以 .com 为扩展名的全部命令文件；包括以 .exe 为扩展名的各种可执行文件，这一点 dBASE III 是办不到的。

例如：

1. 显示磁盘目录，可键入：

+> // dir 或 +> // dir/w 或 +> // dir <文件名>

2. 列出磁盘文件的内容，可键入：

+> // type <盘符:文件名.扩展名>

3. 采用行编辑来写 db4 的命令文件时，可键入：

+> // edlin <盘符:文件名.db4>

4. 采用 BASIC 语言编写并编译后的某个文件 BASAIC.exe，可以嵌套在 db4 的命令文件中，如部分程序流程为：

db4 命令文件语句序列

```

//BASIC.exe      (转去执行 BASIC 语言的文件，当执行完毕后继续
                  执行 db4 的命令序列，直到结束)

```

end

(E) db4 的数据文件可转换成扩展名为 .TXT 的 ASCII 码文本文件，从而被其它高级语言所读写；同样，其它高级语言或字处理程序所建立的带 .TXT 扩展名的文本文件，也可转换成 db4 的数据文件，只需要注意保持数据结构约定的一致性即可。当然，dBASE II 和 dBASE III 也具有这种功能，但在一定条件下，db4 可直接用字处理程序来编辑，并用高级语言来读取。如由 edlin 编辑的 db4 数据文件，可由 BASIC 的 input 语句来顺序地读取 db4 的数据文件中的数据。这种相互读取的关系，给 db4 提供了有力的手段，使 db4 去改造非 db4 的管理软件，使其更加合理和灵活，是十分有益的。与高级语言结合起来，克服了 dBASE 的不足，发挥各自的优点，从而促进软件开发水平的提高。

(四) db4 简便易学，总共只有 44 条命令，而 dBASE III 却有 117 条命令，db4 的若干命令具有多种功能，只要组合或任选得当，其功能是十分灵活的。并且，其命令的关键字多近乎英文原意，便于记忆。尽管命令条数比 dBASE III 少了几近一半，但功能却差不多，而且还增加了一些新的命令，例如：

1. 输入变量的命令 input 能够做到：

(1) 输入数值型变量，这就相当于 dBASE III 的 input 命令。

(2) 输入字符型变量，这就相当于 dBASE III 的 ACCEPT 命令。

2. 在屏幕上显示并修改数据文件的内容时所用的 **screen** 命令, 能够做到:

(1) 竖向显示一条记录, 这就相当于 **dBASE III** 的 **edit** 命令。

(2) 横向(加 **HF**)显示一条记录, 这就相当于 **dBASE III** 的 **BROWSE** 命令。

3. 与 **dBASE III** 相同的命令, 其功能大有扩充。如 **append record** 可把同一数据文件中任一条记录的内容, 追加到该文件的最后一条记录中去, 这样就避免了若干相同数据的重复输入工作。

4. 可在两个数据文件的某两条记录之间相互传送数据, 只要字段名相同就可以, 如:

+>open #1 file1 !打开1号文件file1。

+>open #2 file2 !打开2号文件file2。

+>read #1 10 !将1号文件file1的记录指针指向第十条记录。

+>read #2 20 !将2号文件file2的记录指针指向第二十条记录。

+>move #1 to #2 !将1号文件file1的第十条记录的内容传送到2号文件
!file2的第二十条记录中。

(E) **db4** 命令语句的长度为 512 个字节, 而 **dBASE III** 为 256 个字节。**db4** 可以连续六次续行, 这样就增加了语句的功能。

(F) **db4** 的数据库技术与高级语言可以相互结合, 融为一体, 所以, **db4** 有许多高级语言性质的命令, 例如:

1. **gosub—return** 命令, 类似于 **BASIC** 语言的命令。但它增加了对输入数据正确性判断的功能, 即 **return** 返回命令, 一般有三种功能。

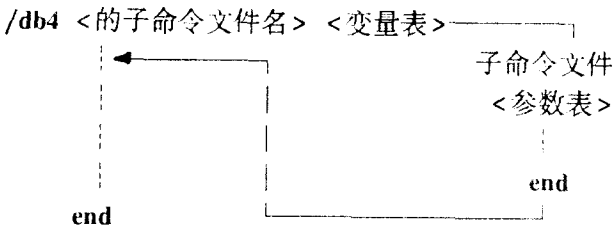
(1) **return(1)**: 正常返回, 执行下一条语句。

(2) **return(2)**: 返回原调用处, 并重新输入数据后, 再进行调用。

(3) **return(3)**: 停止数据的输入, 并继续向下执行。

2. 通过执行 **db4** 的 **/** 命令来调用子命令文件, 该子命令文件可带参数表, 相当于高级语言中的有参过程, 将主命令文件中的变量传送到子命令文件相对应的参数中, 如下图:

db4命令文件主程序



3. **db4** 的命令文件中可用 **GOTO <标号>** 语句, 以转移到标号指定的位置处继续执行命令, 这样就能从 **if-endif** 程序段中跳出来, 从而提高了简化程序的手段。

(G) **db4** 的内存变量个数为 500 个, 而 **dBASE III** 是 256 个, 且数学函数比 **dBASE III** 多, 如正切函数 **tan()**; 反正切函数 **atn()**; 正弦函数 **sin()**; 余弦函数 **cos()**; 平方根函数 **sqr()**; 自然指数函数 **exp()**; 十进制转十六进制函数 **hex()**。可进行较复杂的科学计算。另外, 字符串型数据运算函数还增加有左取字符 **left\$()**; 右取字符 **right\$()**; 中取字符 **mid\$()**; 取首字符 **string\$()** 等。在运算符中还增加了直接乘方([^])的运算符。所以它近乎乎 **BASIC** 语句的运算功能, 这一点比 **dBASE III** 要优越。

(H) 能连接多个数据库文件, 可同时打开九个 **db4** 的数据文件, 并规定其各自的文件

号。在对字段名进行操作时，只需指明其所在文件号即可。例如，把第五号文件的第五条记录中的第一个字段 file51 乘以第六号文件的第六条记录中的第二个字段 file62，再加上第七号文件的第七条记录中的第三个字段 file73，其结果写入第五号文件的第五条记录中的第二个字段 file52 中，其程序为：

```
open #5 file5      !打开五号数据文件。
open #6 file6      !打开六号数据文件。
open #7 file7      !打开七号数据文件。
read #5 5          !把五号文件的记录指针指向第五条记录。
read #6 7          !把六号文件的记录指针指向第六条记录。
read #7 7          !把七号文件的记录指针指向第七条记录。
#5.file52=#5.file51*#6.file62+#7.file73
                  !赋值运算。
write #5           !将结果写入五号数据文件中。
? loc(7)           !使用loc( )函数来测试七号文件的当前记录号。
7                 (测得的结果是第七条记录，说明原有的指针并没有移动)
```

由此例显而易见，对于比较复杂的程序设计和检查、维护，在 db4 中是简单明了的，与 dBASE III 相比，在多个数据库关联使用时，db4 要方便、灵活、实用些。

1-2 db4 的环境。

(一) 硬件环境：

1. 微机系统：IBM PC/XT；IBM PC/AT；IBM 5550；GW05 系列等及其兼容机。
2. 内存容量：大于 256KB，运行“王码”汉字时大于等于 512KB。
3. 外存容量：由操作系统支持的软盘和硬盘驱动器所确定，最少应有一个软盘驱动器。
4. 显示器：中分辨率以上的显示器。
5. 外设：至少有一台 80 列以上的打印机。

(二) 软件环境：操作系统为 MS-DOS 2.0 或 PC-DOS 2.0 以上的版本；运行汉字时要有 CCDOS 2.0 以上的中文操作系统。

1-3 db4 的结构和性能。

(一) db4 的程序结构包括五个文件，它们是：

1. command.com: 程序长度为 17.792KB，这是一个操作系统命令管理程序文件，其版本应与 DOS 一致，运行 db4.exe 时必不可少。
2. db4.exe: 程序长度为 115.792KB，这是一个 db4 的执行程序文件，使用时一次调入内存，这样在用户执行程序时，不再需要调入其它的.exe 文件。
3. DBED.exe: 程序长度为 20.088KB，这是一个编辑命令文件的程序，即 db4 编辑器，用来建立一个新的.db4 文本文件，或用来修改，增删，编辑一个原有的文本文件，它在 DOS 下单独运行。

4. DBSCR.exe: 程序长度为 35.328KB, 用来建立输入和输出屏幕格式的程序文件, 简称 DB 屏, 它可在屏幕上的任何地方画线, 画图, 写标题等。

5. DBCNV.exe: 程序长度为 17.766KB, 用来将 dBASE II 数据文件转换成 db4 数据文件, 在 DOS 下单独运行。

(C) db4 性能指标及其与 dBASE II, dBASE III 的比较。

性 能	db4	dBASE II	dBASE III
数据文件记录的条数	最多 65535 条	最多 65535 条	最多一千兆条
每条记录的字符数	最多 4000 个	最多 1000 个	最多 4000 个
每条记录的字段数	最多 64 个	最多 32 个	最多 128 个
可同时打开的文件数	最多 10 个	最多 2 个	10 个工作区
同时打开观察文件数	最多 9 个		最多 7 个
字段类型种类	最多 3 种	最多 3 种	最多 5 种

性 能	db4	dBASE II	dBASE III
内存变量个数	最多 500 个	最多 64 个	最多 256 个
有效数值的位数	BCD 码 14 位		19 位
字符串长度	最多 254 个		最多 254 个
索引关键字长度	最多 80 个		最多 100 个
文件名长度	最多 8 个	最多 8 个	最多 8 个
字段名长度	最多 10 个		最多 10 个
命令语句长度(字符)	最多 512 个		最多 256 个
算术运算精度	最多 6 位		
命令条数	44 条		117 条
函数个数	35 个		37 个

1-4 db4 的操作方式。

一旦进入 db4 状态, 屏幕上出现 db4 的提示符 +>, 说明当前处于 db4 的命令级, 就可以执行 db4 的操作, 其操作方式有两种。

(一) 直接命令方式: 在 db4 命令级状态下, 按 db4 的语法输入命令后, 一按回车就立即执行该条命令。多数情况下是执行单条语句或人机对话时采用。

(二) 程序方式: 在 db4 命令级状态下, 按 db4 的命令文件来执行。

1. 由"/"斜线开始执行 db4 命令文件。

2. db4 的命令文件可用字处理 WS; 行编辑 EDLIN; 程序编辑器 DBED.exe 等等工具软件来编制命令文件, 命令文件必须采用.db4 的扩展名。

1-5 db4 的语句。

语句是 db4 执行过程中的最小描述单元, 它包括有命令名, 文件名, 条件短语, 即表达式和作用范围等等, 不同命令的组合是各不相同的, 命令的格式与其功能将在以后分别介绍。

(一) 语句的一般格式。

1. <命令字> <#文件号> [<表达式表>] [记录限制范围]

- (1) < > 中的内容由用户给定, 是必用项。
- (2) [] 中的内容是可选项, 用来扩充语句的功能, 是可有可无的。
- (3) 命令字(即命令名): 是 db4 的命令, 指明进行何种操作。
- (4) 文件号: 指定被操作的文件是哪个号, 在打开文件时指定该号, 如
+>open #1 file1 !打开数据文件 file1, 并指定其为 1 号。
- (5) 表达式表: 是一组由逗号分隔开的表达式。
 - ① 当表达式为字段名时, 表示只对记录中的该字段进行操作。
 - ② 当没有表达式时, 即表达式的默认值, 表示对记录的全部字段进行操作。
- (6) 记录限制范围: 表示由 where 或 while 条件短语限制的处理记录的范围。

2. 语句的一般格式可进一步写成:

<命令字> [<#文件号>] [<范围>] [[while<表达式>] [where<表达式>]]...

例如: relate [<文件号 1>] [<范围>] [[while<表达式>]

[where<表达式>] [{attribute:att}<字段名表>] [join

<文件序 3>] [关键字=<表达式>,...<文件号 n>

[关键字=<表达式>] to {<新文件名>:<文件号 2>}]

[text [dlm] "<定界符>"] [gosub <子程序名>]

其中{ : }表示冒号前后任选一种。这条语句表明了命令的结合功能是很强的。

(二) 语句的长度及续行。

1. db4 语句的最大长度是 512 个字符, 比 dBASE III 的语句长度 256 个字符长出一倍。而一行中只能容纳 255 个字符, 超出部分就要续行书写。每一行最多为 255 个字符。

2. 续行的办法有下列几种。

- (1) 在行尾用"/"的办法: 在"/"后从第一个非空格开始的字符是续行后的字符, 即:

xxxxxxxxx/

-----ON CODE to cosT

└ 空格 ┘ └ 续行字符 ┘

续行从 on 开始, on 之前的空格不做为续行字符。

- (2) 在行尾用"/;"的办法: 在"/;"后包括空格在内之全部字符均做为续行字符, 即:

xxxxxxxxxxxxx/;

-----on CODE to cosT

└ 续行字符 ┘

续行从第一个空格开始。

- (3) 在行尾用"/", 同时在行首用";"的办法: 紧跟在分号后的字符均为续行字符, 包括空格在内。即:

xxxxxxxxxxxxx/

;-----on CODE to cosT

└ 续行字符 ┘

- (4) 例如:

① 第 n 行为 +>tatol/

② 第 n+1 行为 +>; where cosT>1000/; !注意, 在 w 前有一个空格应包

!括在续行中。

- ③ 第 $n+2$ 行为 `+> on CODE to TcosT` !注意, 在 O 前有一个空格应
!包括在续行中。

上述三行的等效行为 `+>tatol where cosT>1000 on CODE to TcosT`

- (5) 可利用宏代换&符号来续行, 例如:

```
+>AA$="all to"  
+>BB$="SALES"  
+>relate &AA$/  
+>    &BB$
```

以上等效于 `+>relate all to SALES`

- (6) 在 db4 中续行最多只能使用六次。

3. 在直接执行方式中, 也可用上述办法进行续行。

- (c) 直接执行方式中语句的修改。

利用 DOS 中的功能键, 可对 db4 的直接执行方式语句进行修改和编辑, 而无需重新输入语句。因为键入的字符串, 开始是放在“输入缓冲区”中的, 利用功能键将语句调到命令行, 再对其进行修改即可。

1. 标准功能键在不同的操作系统中的规定有所不同, 在 MS-DOS 操作系统中的主要功能键有如下几种:

- (1) CI (F1): 按下此键, 从原行(样板)中复制一个字符到新行中, 并显示该字符。
- (2) CV (F2): 按下此键, 同时输入一个字符, 则从原行(样板)中复制字符, 直到遇到指定的字符为止。
- (3) CA (F3): 按下此键, 把样板(原行)上所有其余的字符全部复制到屏幕上。
- (4) SI (del): 按下此键, 删除掉样板(原行)中光标所在的那的一个字符。
- (5) SV (F4): 按下此键, 并输入一个字符, 则删除样板(原行)中该字符前所有的字符, 直到遇到被指定的字符为止。它与 F2 键的作用相反。
- (6) VOID (ESC): 删除刚输入的内容, 而不改变样板(原行)中的内容。即输入的字符作废。

- (7) NWL (F5): 把现行行的内容移到样板(原行)行中去。

- (8) INS: 按下此键, 允许在一行中插入内容。再按下此键, 则退插入。

2. 现以实例说明其用法。欲要输入下列语句:

```
+>relate #1 all where #1.CCODE="010"/;  
+> join #2.key=#1.CCODE,#3 key=#1.PCODE/;  
+> att #2.CNAME,#3.PNAME,#1.MMDDYY,#1.QTY to SDATA
```

但是, 错误的输入成下列语句

```
+>relate #1 all where #1.CCODE="010"/;  
+> join #2 KET=#1.CCODE #3 key=#1.PCODE/;  
+> att #3.PNAME,#1.MMDDYY,#1.QTY to SDATA
```

对此, 修改的步骤如下:

- (1) 由于第一行没有出错, 所以照原样复制, 按下标准功能键符 F3, 则第一行将显示出来为:

+>relate #1 all where #1.CCODE="010"/;

(2) 按回车 enter, 结束第一行的修改, 并在下一行上出现提示符+>

(3) 将第二行的 KET 改为 key, 首先, 键入标准功能键 F2, 再键入 T(注意, 大小写要与原字符一致), 则将 T 前的部分字符显示出来如下:

+>relate #1 all where #1.CCODE="010"/;

+> join #2 KE

(4) 键入改正的字符 Y, 然后再键入原样复制的功能键 F3, 则第二行余下的部分将显示出来如下:

+>relate #1 all where #1.CCODE="010"/;

+> join #2 key=#1.CCODE,#3 key=#1.PCODE/;

(5) 按回车 enter, 结束第二行的修改, 并在下一行显示提示符+>。

(6) 在第三行 att 之后, 丢掉了 #2.CNAME, 现予加上。先原样复制四个字符 att, 即首先键入复制单个字符的功能键 F1 四次, 就出现了 att。

(7) 接着键入功能键 INS, 进入插入方式。

(8) 键入丢掉的 #2.CNAME。

(9) 键入原样复制的功能键 F3, 则将第三行其余部分显示出来, 至此全部修改完毕, 与原要求的语句一样。

+> att #3.PNAME,#1.MMDDYY,#1.QTY to SDATA

(10) 由于第三行尾部没有/, 所以按回车 enter 就执行这条语句。

(四) db4 的保留字: 保留字有以下几种。

1. db4 的命令字: 语句通常以命令字开始, 命令字是语句的核心部分, 是一个英语动词。它表示了语句要执行的操作, 如定义数据结构的 define; 打开数据文件的 open; 显示数据文件内容的 display 等等。

2. db4 的短语引导词: 为了把命令的动作描述得确切, 常常要附加一些短语。如记录限制范围用的 while, where; 指定记录用的 record; 表示属性用的 attribute 等等。这些都是短语引导词。

3. 为了对数据进行运算, 还有若干函数。如求绝对值的函数 abs; 求反正切的函数 atn; 求记录号的函数 loc; 置空格用的函数 space 等等。

4. 命令字, 短语引导词和函数的名字等均为 db4 数据库管理系统所专用, 均有其特定的功能和用法, 统称为保留字, 不可作为它用。即不能用保留字做为文件名, 字段名, 变量名和标志名等等。

5. db4 中所用保留字, 按字母顺序排列如下:

all	defseg	key	recall	xor	peek
and	descending	keyin	record	abs	right\$
append	display	kill	relate	asc	sgn
ascending	dln	kin	rename	atn	sin
attribute	do	locate	report	cap\$	space\$
beep	drive	message	return	chr\$	sqr
beginwp	echo	mod	screen	cos	str\$
blank	eject	modify	sort	date\$	string\$

bottom	else	move	stop	del	tan
bracket	end	msg	structure	exp	trim\$
break	endcase	next	subroutine	file	time\$
call	enddo	off	sum	eof	val
carry	endif	on	system	fix	view
case	endwp	not	take	hex\$	
change	files	open	talk	instr	
chg	find	OR	text	int	
clear	form	otherwise	to	left\$	
close	format	parameter	top	len	
cls	get	pass	total	loc	
console	give	picture	while	lof	
count	gosub	poke	whl	log	
define	input	print	whr	match	
delete	join	read	write	mid\$	

6. 保留字的缩写形式:

(1) 在一般情况下, 取保留字左端的三个或四个字符为其缩写形式, 只要能区别开与其它保留字的不同即可。如:

display 缩写成 dis或disp.

structure 缩写成 str或stru。

open 缩写成 ope。

(2) 当某些保留字左端三个字符相同时, 则取左端的四个字符做为其缩写形式, 如:

endcase 缩写成 endc。

enddo 缩写成 endd。

record 缩写成 reco。

recall 缩写成 reca。

(3) 下列七个特殊的保留字, 其缩写形式如下:

change 缩写成 chg或cha。 print 缩写成 ?。

keyin 缩写成 kin。 where 缩写成 whr。

locate 缩写成 @。 while 缩写成 whl。

message 缩写成 msg。

(4) 以 F 或 f 打头, 后随两位数字者, 做为保留字处理。如 F01, f12, F03\$, F14\$等等。

(5) 35 个函数的保留字, 不能采用缩写形式。

1-6 db4 的文件。

向数据库中输入的各种数据和程序, 均以文件的形式存放在磁盘中。每一个二维数据的关系便是一个数据库主文件。所以, 文件是一组有名字的信息的集合。在 db4 中, 按不同用途, 把文件共分为五类。即命令文件、数据文件、观察文件、报告文件和文本文