一个用Turbo-C实现的软件集成器

唐常杰　韩仲清　　四川大学计算机系

摘要：本文提供的用Turbo-C实现的软件集成器能把在物理上分散的软件，从逻辑上集成于一个统一的界面之下。采用控制与数据分离的办法，用户不必修改和重编译C程序，而只需用文字处理器修改菜单文本文件和一个批处理文件，就能适应用户的特殊需要。

关键词：Turbo-C　软件集成器

在软件开发过程中，有时需要将分散在不同驱动器、不同目录下的各种软件集中在一起。本文提供的用Turbo-C实现的软件集成器就具有这一功能。

软件集成器共占磁盘空间约6k字节，由三个部分组成：

收稿日期：1991年3月11日

```
 98: SEND:    fclose(stream);
 99:     fin:HSETBOX(20*8-8,10*8-8,60*8+4,20*8+4,0);
100:        HPTMP(qq1,3);
101:          }
102:       /*-------------------------------;
103:       ;FUNCTION:get scan code of a key ;
104:       ;RETURN: scan code               ;
105:       ;-------------------------------*/
106:       GETKEY( )
107:             {
108:             inregs.h.ah=0;
109:             int86(0x16,&inregs,&outregs);
110:             return(outregs.h.ah);
111:              }
112: /*---------------------------
113: ;FUNCTION:clear state windows;
114: ;         and location.     ;
115: ;---------------------------*/
116:   STATECL( )
117:         {
118:         HSETBOX(79,184,600,192,0);
119:         LOCATE(23,10,0);
120:          }
121:
```

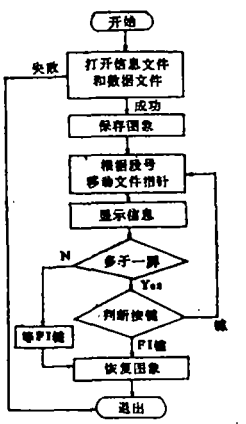


图1　主程序流程图

3　结束语

本程序为应用程序使用Help功能提供了良好的接口，是Help功能理想的开发工具。我们已成功地将此运用到我们所开发的多个应用软件中。

（限于篇幅，本文只将主程序附后供参考。若对汇编的五个子程序感兴趣，请与本编辑部联系）

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

1 选择器

这即是由图1的C程序Select.C编译而成的Select.EXE。其中函数Open Menu Text打开菜单文本文件备读。程序中第18行检查Select.EXE的命令行参数，如有参数，则用参数作输入文件（因此用户可以指定其它菜单文件名），若无参数，则默认菜单文件名为Menu。第23行打开文件，第27行显示菜单，第28～35行进行选择。当用户键入"Q"或"q"时，令Choice为128，用户敲回车键时，Choice取默认值1，第36行以用户选择的Choice值作为退出码。程序退出后返回到父进程中。至少有三种方法可以按退出码分流：

①父进程为Turbo pascal程序，（用过程EXEC运行Select.EXE），则可用"Case ExitCode of"语句分流。

②父进程为Turbo-C程序，用Spawn函数族运行Select.EXE，可用Spawn函数族的返回值进行分流处理。

③父进程为DOS的Command.COM，可用"if Errorlevel n goto…"进行分流处理。

本软件集成器采用第3种方法。

```
/* 文件名  : Select.C */
#include <stdio.h>                    /* 2 */
#include <stdlib.h>
#include <conio.h>
#include <ctype.h>
#include <string.h>                   /* 6 */
FILE *fp;  /* 菜单文本文件变量 */
/*-----------------------------------*/
void  DisplayMenu( )                  /* 9 */
 /* 显示菜单文本文件,设 fp 已经打开 */
  {  char ch;
     while( (ch=getc(fp)) != EOF )
     putchar(ch);  }
/*-----------------------------------*/
main(int argc, char * argv[] ) /* 15 */
 { char *FileName, *choiceStr;
   int choiceNum;
   if (argc>=2 ) FileName=argv[1];
       /* 有参数则用参数作输入文件 */
   else FileName="Menu";
      /* 无参数则用默认文件名 */
  /* 打开文件备读 :  */                 /* 22 */
   fp = (FILE *) fopen(FileName,"rt");
   if (fp==0)
   { printf(" Can not open MENU. \n");
     exit ( 128 ); }                  /* 26 */
   DisplayMenu( );  /* 显示菜单 */
   choiceStr=(char *) malloc(80);
   do { gets(choiceStr);          /* 29 */
        if (strlen(choiceStr)==0)
        choiceStr="1";
   else if (toupper(choiceStr[0])=='Q')
        choiceStr="128";
        choiceNum = atoi(choiceStr);}
   while ( choiceNum == 0 );
   exit(choiceNum); }          /* 36 */
```

图 1

2 菜单文本文件

菜单文本文件的默认名称为Menu。用户可以用任何文字处理器（如Wordstar）编辑，亦可加注解，甚至汉字注解。图2给出了一个例子，其中选择1、2分别启动dBASE、Wordstar，键入Q则退出。这里对被集成软件的个数没有限制，用户可根据需要修改。注意：文件的结束应在图2中标出的<EOF>处，以后显示菜单时光标才能在该处闪动。

文件名：Menu

```
                Menu
        ----------------------
              1. dBase
              2. WordStar
              Q. Quit
        ----------------------
        choice [1-3,Q] 1<EOF>
```

图 2

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

用汉字BASIC语言实现全屏幕数据编辑

姚国祥　　武汉工业大学管理系

摘要： BASIC语言具有灵活、易学、会话和绘图功能强等特点，可它在全屏幕数据编辑，即同时对多个数据进行修改却显得无能为力，从而大大抑制了BASIC语言的应用，本文介绍一个通用子程序，它可以模拟dBASE语言中的全屏幕数据编辑功能，弥补了BASIC语言的这一欠缺。

关键词： BASIC语言　数据编辑　屏幕显示　全屏编辑

1　问题的提出

使用dBASE语言进行数据编辑（包括输入数据和修改数据）非常方便，不但可以在屏幕定义一个很美观的格式，而且在同一屏幕上能编辑多个数据，即可使用全屏幕数据编辑，这给数据的输入，尤其是数据的修改带来了极大的方便，这也是人们乐于使用dBASE语言的重要原因之一。然而，BASIC语言虽然有很强的人机对话功能，可是在进

收稿日期：1990年12月7日

3　用于集成分流的批理处文件Integrit.BAT

批处理文件Integrit.BAT如图3所示，括号中的行号是为了说明方便而加的，录入时连同括号一起去掉。为了保证集成界面的统治地位，第4行用了循环标号loop。在主菜单下运行任何软件，只要一退出该软件就又回到主菜单。第9行Select Menu语句中完成菜单显示和选择。Menu是Select的默认菜单文本文件名，可以省略，用户也可以用其它名称来取代Menu。用户的选择将表现为Errorlevel值（出错层号值）。

由于高层错误必然也是低层错误，因此在第13～17行用IF语句分流时，必须把出错层号大的排在前面。如果把第17行与第14行对调，则无论用户作什么选择，返回的出错层号码总大于等于1，在检查IF Errorlevel 1时总为真，因而都为goto 1，于是便不能运行其它选择项。

我们在实验室的PC/XT的AutoExec.BAT中嵌入了该软件集成器，把20兆硬盘上的35个软件集成在一个界面下，使用十分方便，取得很好的集成效果。

```
Rem 文件名：Integrit.BAT.
echo off                          (1)
CLS
GOTO START
:loop
  CLS                             (5)
  PAUSE
:START
CLS
REM 显示菜单和选择
Select Menu                       (10)
REM 用 Errorlevel 检查选择码：
REM 大的放前面，按 Q 为 128
IF ERRORLEVEL 128 GOTO end
IF ERRORLEVEL 2 GOTO 2
IF ERRORLEVEL 1 GOTO 1
 :1                               (15)
   cd C:\dbase
   dbase
   GOTO loop
 :2                               (20)
   cd C:\WordStar
   ws
   GOTO loop
 :End
 ECHO       Good bye !
                                  (25)
```

图　3

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

行数据输入时却远不能象dBASE语言那样随心所欲；BASIC用来输入数据的常用语句是INPUT，它不但没有美观的格式，而且不能利用原来的数据进行修改，这给使用者带来了很多麻烦，本人设计了一个BASIC通用子程序，来模拟dBASE语言的全屏幕编辑功能，收到了较好的效果。

2 功能的构造

模拟全屏幕数据编辑，用小键盘控制光标的移动，如→键向后移动一个字符，←键向前移动一个字符，↑键向上移动一个数据（或数据行），↓键向下移动一个数据或数据行，回车键结束一个数据的输入并将光标移到下个数据的起始位置；Home键可将光标直接移至第1个编辑数据的起始位置；End键将光标直接移至最后一个编辑数据的位置；Ins键插入字符，Del键删除一个字符。

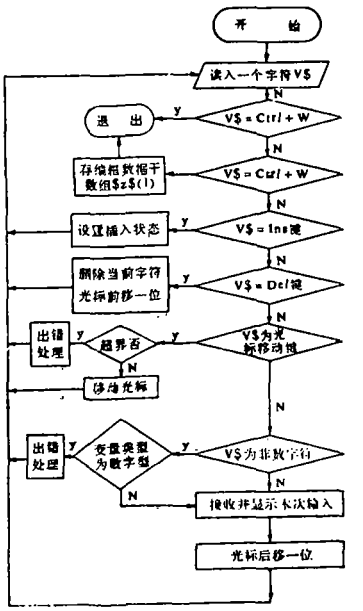
另外，对输入的命令键或字符要有识别和出错处理功能，如光标的越位，数据类型的输入错误等等，如在编辑数字型变量时输入了字母或其它字符，则程序发出“嘟”的声音以示警告，并拒绝接收所输入的字符。

在编辑完最后一个数据时，程序自动退出，所有数据的编辑（或修改）有效；在数据的任何编辑过程中，也可以中途退出，按Ctrl+w键为有效退出，即所有编辑过的数据有效，并替换原有数据变量的内容，按Ctrl+Q键为无效退出，即所有编辑的数据无效，变量原先的内容被继续保留，亦称为放弃本次编辑。

这些功能均仿制于dBASE语言的全屏幕数据编辑功能，为用户进行数据输入或修改提供了友好的人机界面。

3 实现的原理

在高级BASIC语言中有一个函数INKEy$，它的功能是从键盘上读入一个字符这给我们实现全屏幕编辑提供了可能，利用它，可以从键盘上读入一个数字串，然后把这个数字串转换为数字；同样也可以读入一个字符串，从而完成字符变量的输入；利用INKEy$函数，也可以读入一个命令键，如改变光标的位置，和LOCATE语句配合使用，可以很方便地移动光标到屏幕的任意位置，从而实现对任意多个变量的数据编辑，读入数据的流程图如下图所示：



数据编辑流程图

根据上述原理和流程图，便可编制一个全屏幕数据编辑的通用子程序，详见程序清单中的3000～3305语句。在调用这个子程序之前，必须在主程序中对数组sz$、YAO%及变量MAX定义并赋值，数组sz$存放所编辑的数据（或需要修改的数据），这个一维数组的大小由所需编辑的数据个数而定。数组YAO%是个二维数组，存放待编辑数据的特性参数，它详细地描述了待编辑数据变量的类型、长度、在屏幕上的坐标位置。具体地说就是：YAO%(i, 1)表示第i个数据的类型，其中1表示数字型，2表示字符型；YAO%(i, 2)表示第i个数据的总长度；YAO%(i, 3)和YAO%(i, 4)为第i个数据编辑时的屏幕位置，即YAO%(i, 3)

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

表示行坐标，YAO%（i，4）表示列坐标，因此每个待编辑数据都必须确定这 4 个参数。

在这个子程序中，输入功能由下列语句实现：

```
3050  V$=INKEY$: IFV$="" THEN 3050  ELSE VP=ASC(V$)
```

它一直等待输入，当用户按一个键时，便对它进行分析，判断它是数字键、字符键，还是控制键，这种分析比较，不是直接与字符比较，而是比较它的ASCII编码，通过分析输入键值的ASCII编码，从而进行相应的处理。当用户键入错误的操作时，程序通过Beep 语句发出“嘟嘟”报警声，同时放弃本次输入，然后重新进入等待输入状态。

4　一个数据编辑的例子

这里举一个利用上述子程序进行全屏幕数据编辑的例子，需要编辑的共有 9 个数据，它们是：TITLE$, SER1TITLE$, SER2TITLE$, SER3TITLE$, XTITLE$, YTITLE$, YMIN, YMAX, YSTEP，这里既有数字型变量，也有字符型变量，为了处理上的方便，先将数字型变量当作字符型变量处理，然后通过类型变换将它还原成数字型变量。

在调用子程序子前，要做以下四件事：

1．将屏幕置成高分辨图形方式，以便画线，并将屏幕编辑数据的提示的信息布置在适当的位置，并画上格子线，以设计一个美观大方的屏幕编辑格式。

2．定义数组YAO%（10.4），并根据其参数含义给予赋值，即各数据变量的类型，数据长度，变量编辑的行列坐标。

3．定义数组sz$(9)作为数据缓冲，并将需要编辑数据变量的原来内容送入这个缓冲数组中，以待进行编辑，或对它们进行修改。

4．将数据编辑所用的控制命令，显示在屏幕的底部，以便用户操作时参照，以防止误操作。

在做好上述准备工作之后，便可以调用这个全屏幕数据编辑通用子程序了，运行结果如下：

*** * * * * 线条图定义 * * * * ***

整个图形总标题（40）湘乡水泥厂主要指标分析图	
数据系列1标题（10）　总产量	
数据系列2标题（10）　总产值	
数据系列3标题（10）　利润	
X轴标题（10）　月份	Y轴标题（10）数量
Y轴最小值(10)　5000	Y轴最大值（10）9000
Y轴步长（10）　500	回车键盘退出

退出：Ctrl+w，放弃：Ctrl+Q，编辑键：↑，↓，→，←，Home，End

在返回主程序时，因为所编辑的数据还在缓冲数组 sz$ 中，所以应将它们逐个回送到原来的变量，从而实现了对这些变量的数据编辑。

5　小结

这个数据编辑程序是一个通用子程序，可以实现在BASIC状态下对任意数据变量的全屏幕编辑，弥补了BASIC语言在这方面的欠缺，增强了使用BASIC语言编制软件的人机界面的友好性，为用户输入数据和修改数据提供了极大的方便，本程序已在IBM—PC/XT机上调试通过，编译后运行效率比较高，效果也比较好，有兴趣者不妨可以试一试。

程序清单附后

参考文献（略）

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

```
1000 '********** 数据编辑主程序 **********'
1005 DIM SZ$(10),YAO%(10,4)
1010 SCREEN 2:CLS :KEY OFF
1015 LOCATE 2,16 :PRINT "* * * * * * 线 条 图 定 义 * * * * * *"
1020 PRINT TAB(10);"整个图形总标题 (40)"
1025 PRINT TAB(10);"数据系列1标题 (10)"
1030 PRINT TAB(10);"数据系列2标题 (10)"
1035 PRINT TAB(10);"数据系列3标题 (10)"
1040 PRINT TAB(10);"X轴标题 (8)"; :PRINT TAB(42);"Y轴标题 (8)"
1045 PRINT TAB(10);"Y轴最小值 (8)"; :PRINT TAB(42);"Y轴最大值 (8)"
1050 PRINT TAB(10);"Y轴步长 (10) "; :PRINT TAB(42);"回车键盘退出 " ;
1055 FOR I=1 TO 9
1060 LINE (60,18*I-1)-(560,18*I-1)

1065 NEXT I
1070 LINE (60,17)-(560,161),,B
1075 LINE (40*8-1,6*18-1)-(40*8-1,9*18-1),,B
1080 YAO%(1,1)=2 :YAO%(1,2)=40 :YAO%(1,3)=3 :YAO%(1,4)=30
1085 YAO%(2,1)=2 :YAO%(2,2)=8 :YAO%(2,3)=4 :YAO%(2,4)=40
1090 YAO%(3,1)=2 :YAO%(3,2)=8 :YAO%(3,3)=5 :YAO%(3,4)=40
1095 YAO%(4,1)=2 :YAO%(4,2)=8 :YAO%(4,3)=6 :YAO%(4,4)=40
1100 YAO%(5,1)=2 :YAO%(5,2)=8 :YAO%(5,3)=7 :YAO%(5,4)=28
1105 YAO%(6,1)=2 :YAO%(6,2)=8 :YAO%(6,3)=7 :YAO%(6,4)=58
1110 YAO%(7,1)=1 :YAO%(7,2)=8 :YAO%(7,3)=8 :YAO%(7,4)=28
1115 YAO%(8,1)=1 :YAO%(8,2)=8 :YAO%(8,3)=8 :YAO%(8,4)=58
1120 YAO%(9,1)=1 :YAO%(9,2)=8 :YAO%(9,3)=9 :YAO%(9,4)=28
1125 YAO%(10,1)=9 :YAO%(10,2)=1 :YAO%(10,3)=9 :YAO%(10,4)=62 :MAX=10
1130 SZ$(1)=TITLE$ :SZ$(2)=SER1TITLE$ :SZ$(3)=SER2TITLE$ :SZ$(4)=SER3TITLE$
1135 SZ$(5)=XTITLE$:SZ$(6)=YTITLE$ :SZ$(7)=YMIN$ :SZ$(8)=YMAX$
1140 SZ$(9)=YSTEP$
1145 LOCATE 10,12 :PRINT "退出 : Ctrl+W, 放弃 : Ctrl+Q, 编辑键 : ";CHR$(24);",";CH
R$(25);",";CHR$(26);",";CHR$(27);",Home,End";
1150 GOSUB 3000
1155 TITLE$=SZ$(1) :SER1TITLE$=SZ$(2) :SER2TITLE$=SZ$(3) :SER3TITLE$=SZ$(4)
1160 XTITLE$=SZ$(5):YTITLE$=SZ$(6) :YMIN$=SZ$(7) :YMAX$=SZ$(8)
1165 YSTEP$=SZ$(9)
1170 END

3000 '********** 数据编辑子程序 **********
3005 'sz$(i)=数据缓冲,yao%(i,1)=类型,yao%(I,2)=长度,yao%(i,3)=行坐标
3010 'yao%(i,4)=列始坐标,max=数据总个数+1,yao(max,j)为存盘认可位置
3015 '编辑键 : 上,下,左,右,Home,End,Ctrl+W,Ctrl+Q (存盘,放弃)
3020 FOR I=1 TO MAX-1
3025 LOCATE YAO%(I,3),YAO%(I,4) :PRINT SZ$(I);
3030 ROW=YAO%(I,3):COL=YAO%(I,4):COL1=YAO%(I,2)+COL-1 :IF ROW=25 THEN ROW=11.E
3035 LINE((COL-1)*8,18*ROW-1)-(COL1*8,18*ROW-1) '光标补线
3040 NEXT I
3045 NUM=1:LOCATE YAO%(1,3),YAO%(1,4) :ROW=YAO%(1,3) :COL=YAO%(1,4)
3050 V$=INKEY$ :IF V$="" THEN 3050 ELSE VP=ASC(V$)
3055 IF VP=17 THEN RETURN ' Ctrl+Q 放弃退出
3060 IF VP=23 THEN 3265 ' Ctrl+W 读数退出
3065 IF NUM<>MAX THEN 3075 ELSE IF VP=0 THEN VP=ASC(MID$(V$,2)) :GOTO 3165
3070 IF VP<>13 THEN BEEP :GOTO 3050 ELSE 3265 '读数退出
```

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

```
3075 IF VP=13 THEN INS=0 :GOTO 3145 '回车
3080 IF VP<>0 THEN 3200 ELSE VP=ASC(MID$(V$,2)) '控制键
3085 IF VP<>82 THEN 3100 'Ins
3090 IF INS=1 THEN INS=0 ELSE INS=1
3095 GOTO 3050
3100 IF VP=80 OR VP=77 OR VP=72 OR VP=75 OR VP=71 OR VP=79 THEN INS=0
3105 IF VP<>83 THEN 3130 'Del
3110 COL=POS(0) :OP$="" :ROW=YAO%(NUM,3) :IF ROW=25 THEN ROW=11.1
3115 FOR I=COL+1 TO YAO%(NUM,4)+YAO%(NUM,2)-1 :OP$=OP$+CHR$(SCREEN(YAO%(NUM,3),I))
   NEXT I
3120 OP$=OP$+" " :LOCATE YAO%(NUM,3),COL :PRINT OP$; :COL1=POS(0)-1 :LOCATE ,COL
3125 LINE((COL-1)*8,18*ROW-1)-(COL1*8,18*ROW-1) :GOTO 3050 '光标补线
3130 IF VP=71 THEN LOCATE YAO%(1,3),YAO%(1,4) :NUM=1 :GOTO 3050 'Home
3135 IF VP=79 THEN LOCATE YAO%(MAX,3),YAO%(MAX,4) :NUM=MAX :GOTO 3050 'End
3140 IF VP<>80 THEN 3150  'Down
3145 NUM=NUM+1 :LOCATE YAO%(NUM,3),YAO%(NUM,4) :GOTO 3050
3150 IF VP<>77 THEN 3165 'Right
3155 IF POS(0)<YAO%(NUM,4)+YAO%(NUM,2)-1 THEN LOCATE ,POS(0)+1 :GOTO 3050
3160 NUM=NUM+1 :LOCATE YAO%(NUM,3),YAO%(NUM,4) :GOTO 3050
3165 IF VP<>72 THEN 3180 'Up
3170 IF NUM=1 THEN BEEP :GOTO 3050
3175 NUM=NUM-1 :LOCATE YAO%(NUM,3),YAO%(NUM,4) :GOTO 3050
3180 IF VP<>75 THEN 3195  'Left
3185 IF POS(0)>YAO%(NUM,4) THEN LOCATE ,POS(0)-1 :GOTO 3050
3190 IF NUM=1 THEN BEEP :GOTO 3050 ELSE NUM=NUM-1 :LOCATE YAO%(NUM,3),YAO%(NUM,4)
:GOTO 3050
3195 BEEP :GOTO 3050 '非法控制键
3200 IF YAO%(NUM,1)=2 THEN 3210
3205 IF INSTR("0123456789. -+",V$)=0 THEN BEEP :GOTO 3050
3210 ROW=YAO%(NUM,3) :COL=POS(0) :IF ROW=25 THEN ROW=11.1
3215 IF INS=0 THEN PRINT V$;:LINE((COL-1)*8,18*ROW-1)-(COL*8,18*ROW-1) :GOTO 3250
3220 OP$=""
3225 FOR I=COL TO YAO%(NUM,4)+YAO%(NUM,2)-2
3230 OP$=OP$+CHR$(SCREEN(YAO%(NUM,3),I))
3235 NEXT I
3240 OP$=V$+OP$ :PRINT OP$; :COL1=POS(0)-1 :LOCATE ,COL+1
3245 LINE((COL-1)*8,18*ROW-1)-(COL1*8,18*ROW-1)  '光标补线
3250 IF POS(0)<YAO%(NUM,4)+YAO%(NUM,2) THEN 3050
3255 NUM=NUM+1 :LOCATE YAO%(NUM,3),YAO%(NUM,4) :INS=0
3260 GOTO 3050
3265 FOR I=1 TO MAX-1    '读数退出
3270 SZ$(I)=""
3275 FOR J=YAO%(I,4) TO YAO%(I,4)+YAO%(I,2)-1
3280 OP=SCREEN(YAO%(I,3),J) :IF OP=0 THEN OP=32
3285 IF YAO%(I,1)=1 AND CHR$(OP)=" " THEN 3300
3290 SZ$(I)=SZ$(I)+CHR$(OP)
3295 NEXT J
3300 NEXT I
3305 RETURN
```

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net