

## 在 FoxBASE 中一种在线帮助的方法

陈 钰 (国家信息中心数据库部)

在 FoxBASE 应用系统的开发中,如果能够提供一个在线帮助功能,就可以大大方便用户对系统的使用,从而也就提高了应用系统用户界面的友好性。因为屏幕所能提供的解释信息非常有限,往往难于清楚地说明某种功能或信息状态的全部含义,如果系统提供了在线帮助的功能,那么当用户在操作时,遇到了不十分清楚的信息状态,就可以随时查看系统提供的帮助信息,通过这些信息的提示,来作出正确的判断,从而能够准确地进行选择或操作。

另外,在一个应用系统的开发过程中,开发人员通常只是间接地了解用户的某些要求,很难完全理解用户需求的真正含义,并且开发人员对于某些信息所采用的描述方式也不一定马上就能够使用户完全接受。从用户来讲,对于所开发的应用系统也有一个了解熟悉的过程,在这个过程中,只有用户本身才会真正清楚应该在系统的哪些地方,需要有什么样的帮助信息才合适。因此,如果帮助信息能够允许用户自己提供就更理想了,这就需要系统能够提供可由用户自己在线录入和修改帮助信息的功能,这样既可以增强用户对系统的信心和感情,同时系统的用户界面也会更加友好。

### 一、热键与帮助信息的查询

在 FoxBASE 命令中,有一条 ON 命令,它其中的一种使用格式是:

ON KEY = <数值表达式> <命令>

其作用是系统在运行过程中,当某一指定键(热键)被按下时,系统将转移执行 ON 语句中指定的<命令>,在<命令>完成后,返回继续执行发生转移的语句。

通常,我们使用 F1 键作为在线帮助键,由于 F1 的键值为 315,所以 ON 命令的形式为:

ON KEY = 315 DO <帮助程序名>

另外,FoxBASE 中还有一条 SYS 命令,它能为用户返回各种有用的系统值,其中 SYS(18)返回程序正在执行的字段名,也就是热键触发时,程序正在等待用户输入信息的变量名。

于是,为了实现帮助信息的查询,我们首先要建立一个存放帮助信息的数据库 HELPD.B,其结构有两个字段,一个字段存放帮助信息,其字段名为 t\_mess;另一个字段存放帮助信息所对应的变量名,其字段名为 t\_name,并且将该字段作为索引字段。

与查询帮助信息有关的程序分为两部分,一部分为触发热键的程序,而另一部分为查找和显示帮助信息的过程,它们分别如下所示:

#### 1. 与触发热键有关的程序

```
ON KEY = 315 DO HELP.D WITH SYS(18)
```

```
@x1,y1 GET Var1
@x2,y2 GET Var2
```

```
READ
```

#### 2. 查找和显示帮助信息的过程

```
* HELP.D
PROC HELP.D
  PARA fname      &&传递热键触发时的变量名(Vari)
  SAVE Screen     &&保存屏幕示意语句
  SVArea = Select( ) &&保存当前的工作区号
  SELECT 0
  USE Helpdb INDEX Helpdb &&打开帮助信息库和索引
  SEEK fname &&定位到帮助信息库中对应变量的首记录
  DO WHILE .not( ).AND.t_name=fname
    ? t_mess
  SKIP
  ENDDO
  WAIT"按任意键继续"
  USE
  com = "select" + str(SVArea,2,0)
  &com &&恢复原工作区
  RESTORE Screen &&恢复原屏幕示意语句
  RETURN
```

上述程序只是一个示意程序,其中省略了对当时屏幕信息和画面的保存与恢复、帮助信息的格式显示以及帮助信息的翻页控制等细节。

### 二、帮助信息的在线录入

假设我们使用 F2 键作为在线录入帮助信息的热键,那么同样可以把帮助信息的录入程序分为两部分,即一部分为触发热键的程序,而另一部分为帮助信息的录入过程,它们分别示意如下:

## 1. 与触发热键有关的程序

```
ON KEY=316 DO HELP_E WITH SYS(18)
```

```
@x1,y1 GET Var1
@x2,y2 GET Var2
```

```
READ
```

## 2. 帮助信息的录入过程

```
* HELP_E
```

```
PROC HELP_E
```

```
PARA fname &&传递热键触发时的变量名(Vari)
```

```
SAVE Screen &&保存屏幕示意语句
```

```
SVArea=Select( ) &&保存当前的工作区号
```

```
SELECT 0
```

```
USE Helpdb INDEX Helpdb &&打开帮助信息库和索引
```

```
DO WHILE .T.
```

```
temp=""
```

```
ACCEPT "请输入与变量"+fname+"有关的帮助信息" TO temp
```

```
IF LEN(TRIM(temp))=0 &&重复录入,直至输入一空行  
表示结束
```

```
EXIT
```

```
ENDIF
```

```
APPEND BLANK
```

```
REPLACE t_name WITH fname,t_mess WITH temp
```

```
ENDDO
```

```
USE
```

```
com="select"+str(SVArea,2,0)
```

```
&com &&恢复原工作区
```

```
RESTORE Screen &&恢复原屏幕示意语句
```

```
RETURN
```

注意,录入帮助信息使用了 ACCEPT 命令,这是因为调用该过程的当前程序中已经使用了 GET-READ 命令,若在这里再次使用 GET-READ 命令就会发生冲突,FoxBASE 系统将会显示出错警告,并中断程序的运行.除了使用 ACCEPT 命令之外,也可以使用 INKEY 命令来解决录入帮助信息,但它的控制比较复杂。

## 三、帮助信息的在线修改

有了帮助信息的查询和录入程序,就可以把这两个程序结合在一起,组装成帮助信息的在线修改程序.该程序允许用户查询帮助信息,如果觉得不满意,就可以通过按另一个热键(设为 F3)进行修改.为了处理和叙述上的方便,可以假定修改是通过对已有信息进行逐行重新录入来实现的,其程序示意如下:

## 1. 与触发热键有关的程序

```
ON KEY=315 DO HELP_D WITH SYS(18)
```

```
ON KEY=317 DO HELP_M WITH SYS(18)
```

```
@x1,y1 GET Var1
@x2,y2 GET Var2
```

```
READ
```

## 2. 查找和显示帮助信息的过程

```
* HELP_D
```

```
PROC HELP_D
```

```
PARA fname &&传递热键触发时的变量名(Vari)
```

```
SAVE Screen &&保存屏幕示意语句
```

```
SVArea=Select( ) &&保存当前的工作区号
```

```
SELECT 0
```

```
USE Helpdb INDEX Helpdb &&打开帮助信息库和索引
```

```
SEEK fname &&定位到帮助信息库中对应变量的首记录
```

```
DO WHILE .not.eof().AND.t_name=fname
```

```
? t_mess
```

```
SKIP
```

```
ENDDO
```

```
WAIT "按任意键继续"
```

```
USE
```

```
com="select"+str(SVArea,2,0)
```

```
&com &&恢复原工作区
```

```
RESTORE Screen &&恢复原屏幕示意语句
```

```
RETURN
```

## 3. 帮助信息的修改过程

```
* HELP_M
```

```
PROC HELP_M
```

```
PARA fname &&传递热键触发时的变量名(Vari)
```

```
SAVE Screen &&保存屏幕示意语句
```

```
SVArea=Select( ) &&保存当前的工作区号
```

```
SELECT 0
```

```
USE Helpdb INDEX Helpdb &&打开帮助信息库和索引
```

```
SEEK fname&& 定位到帮助信息库中对应变量的首记录
```

```
DO WHILE .not.eof().AND.t_name=fname
```

```
temp=""
```

```
ACCEPT "请修改变量"+fname+"有关的帮助信息" TO temp
```

```
REPLACE t_name WITH fname,t_mess WITH temp
```

```
SKIP &&重复修改,直至最后一行结束
```

```
ENDDO
```

```
USE
```

```
com="select"+str(SVArea,2,0)
```

```
&com &&恢复原工作区
```

```
RESTORE Screen &&恢复原屏幕示意语句
```

```
RETURN
```

过程 HELP\_D 在上述示意中,稍加扩充就能利用上下箭头键把光标定位在帮助信息中需要修改的那一行上,与此同时就可以改造 HELP\_M,使其能够只修改用户指定行的信息了。