

Foxpro 软件联机帮助机制的设计

罗 辉、陈又来 (湖南双峰工商银行)

摘要:本文详细地介绍了 FOXPRO 数据库软件几种上下文相关联机帮助的程序设计技巧。

Foxpro 是 90 年代推出的 FoxBASE+ 换代产品。到目前,它已有三代产品。较之 FoxBASE,它在功能、用户接口、图形窗口、跟踪调试及速度等许多方面都有彻底的改进和全新的完善,使传统的数据库软件业者耳目一新,振奋异常。此外,FOXPRO 还提供了强大的帮助机制,可通过编程以多种方式给用户提供尽可能多的帮助信息,这也是它较之 FoxBASE 的一个重大改进。本文基于 FoxPRO 2.0 / DOS 版本,对其联机帮助机制的设计加以讨论。

一、应用软件中实时帮助信息的设计

Foxpro 提供一个十分有用的微型参考指南,为我们提供了整个软件本身的帮助信息,为获得与程序运行当前操作有关的帮助,必须在此手册上翻阅找到该帮助主题后,才能得到相应的帮助细节。帮助信息不能精确定位,操作也繁琐。有时我们更希望在我们设计的应用软件中提供帮助信息,以便于在程序运行的任意时刻都可获得与正在运行的程序上下文相关的帮助信息。

利用 FOXPRO 众多命令本身可携带帮助信息的功能,可在程序中直接嵌入帮助信息。可携带帮助信息的命令主要有如下:

- 1 (a ...SAY / GET ... MESSAGE '帮助信息'...
——用于在格式化输入 / 输出各编辑字段的信息提示场合。
- 2 (@ ...PROMPT ... MESSAGE '帮助信息'...
——用于光棒菜单上各菜单项的信息显示。
- 3 DEFINE MENU ... MESSAGE '帮助信息'...
——用于对棒式菜单的菜单名提供帮助信息的场合。
- 4 DEFINE POPUP ... MESSAGE '帮助信息'...
——用于对弹出式菜单名提供帮助信息。
- 5 MENU BAR <二维数组>, <数值表达式>
——用于下拉式菜单中菜单棒各选择项的帮助信息提

示。其中二维数组的(i,1)中内容为菜单名,(i,2)内容为相应菜单项的帮助信息。

上述命令的帮助信息提示位置由 set message to <行数> 决定,即显示在行数所指定的那一行上。

用这些命令将可以提供密切的实时帮助信息,光标在不同域上移动时,相应的帮助信息自动在信息行显示,勿需热键激活。但所提供的信息十分有限(最多不能超过一行),其修改也不太方便(必须直接修改程序)。

二、上下文相关联机帮助的设计

FOXPRO 也提供了一些利用帮助库或有助于进行程序上下文相关的联机帮助设计的命令。其提供两条在程序中直接控制帮助主题选择的命令:

SET TOPIC TO <EXPC> 和 SET TOPIC TO <EXPL>

通过它们即可在程序运行中激活 HELP 后自动选择相应的帮助主题并直接将其帮助细节内容显示在 HELP 窗口上。用这种方式提供上下文帮助须用 ON KEY = <热键值> HELP 命令设置 HELP 陷阱,而且只对执行 READ 命令时有效,但 GET 命令中的变量既可以是某数据库字段名也可以是内存变量名。对其它全屏编辑命令如 browse, edit, change 等,仍只提供简易软件帮助手册的帮助方式,并且它们使用的热键由 SET FUNCTION <热键值> TO HELP 设定,而不是由 ON KEY 设定,如果 ON KEY 在 SET FUNCTION TO HELP 命令之后设定而且设定的是同一个热键,那么前者将会屏蔽掉后者的作用,也就是说其后的非 READ 全屏编辑命令中将无法用该热键激活 HELP 窗口获得帮助。编程时必须注意这一点。

1.SET TOPIC TO <EXPC> 命令方式

SET TOPIC <EXPC> 命令是指定 HELP 主题的

最简单方式,在激活 HELP 后,系统首先计算字符型表达式 <EXPC>,然后在由 SET HELP TO 指定的用户帮助库中查找第一个字段(帮助主题)内容与该表达式值相匹配的记录,尤如用 LOCATE 命令定位记录一样。匹配后即在 HELP 窗口显示第二个字段(帮助细节)的内容。用这种方式必须在程序中需要不同帮助时不断地用 SET TOPIC TO 修改主题,以便提供正确的帮助信息。

假设用户帮助库 USERHELP 中有如下记录,其中备注型字段 HELPTTEXT 中为相应主题 TOPIC 的帮助信息:

字段名:TOPIC	HELPTTEXT	FLDNAME	PRGNAME
NAMEHELP	memo	NAME	TMP1
MONEYHELP	meno	MONEY	TMP1
MEMOVARHELP	meno	MEMOVAR	TMP1

且有一个用户数据库 userdata,包括两个字段: NAME, MONEY。则运行下面的小程序:

```
C> TYPE TMP.PRG <CR>
set talk off
set help to userhelp &&指向你的应用程序帮助库
set function 1 to 'help' &&设置 F1 键为 BROWSE 时的
HELP 热键
use userdata
i brow
on key = 315 help &&设置 F1 键为 READ 时的 HELP
热键
memovar = 0 && memovar 是一个内存变量
set topic to 'NAMEHELP'
@ 10,20 say "领域一的值:" get NAME
read
set topic to 'MONEYHELP'
@ 12,20 say "领域二的值:" get MONEY
read
set topic to 'MEMOVARHELP'
@ 14,20 say "变量三的值:" get memovar pict '99999'
read
set help to c: / foxpro2 / foxhelp &&恢复系统帮助库
go top
3 ON KEY &&开放 F1 键为 BROWSE 时的 HELP 热键
2 brow
use
```

这样,当你在 1 处按了 F1 键,可翻阅该系统的联机帮助手册;在 2 处,如果没有 3 处的 ON KEY 命令取消

READ 的 HELP 热键设置,按 F1 键就会毫无反应了;而在 READ 期间,按下 F1 键,此时光标所在域相应的帮助细节将在 HELP 窗口直接显示。

从上面的小程序可以看出,程序中必须含有许多的 SET TOPIC TO 命令,而且不能对同一个 READ 命令内的多个 GET 命令分别提供实时帮助,如果要分别提供帮助则必须用多个 READ 命令——将各个 GET 语句分开,不好进行全屏幕设计,程序冗余量大,不易维护。只适宜用于小型应用软件中。

2.SET TOPIC TO <EXPL> 命令方式

逻辑表达式 EXPL 首先被计算,然后提取使用该表达式为真的用户帮助库第一个记录中的帮助细节内容在 HELP 窗口显示出来。它提供了一种更为通用的设计方式。因为 EXPL 能包括本身属于帮助库的字段名,所以在帮助库中设置其它辅助字段将变得十分有用。可以把任何类型的标识信息放在其中,作为帮助主题的搜索关键值搜寻相应的帮助细节,更有利于上下文相关帮助的设计。

仍以上面的用户帮助库 USERHELP 和数据库 USERDATA 为例,执行下面程序:

```
C> TYPE TMP.PRG
set talk off
set help to userhelp
set function 1 to 'help'
use userdata
browse
memovar = 0
on key = 315 help
set topic to FLDNAME = SYS(18)
@ 10,20 say "领域一的值:" get NAME
@ 12,20 say "领域二的值:" get MONEY
@ 14,20 say "变量三的值:" get memovar pict '99999'
read
on key
set help to c: \ foxpro2 \ foxhelp
use
```

注:因为 SYS(18)返回的 GET 域变量名都是大写,因此在帮助库的 FLDNAME 值也都应大写。

执行上述程序,在 READ 命令执行期间如果你正在输入值,比如输入 NAME 的值,并且你按了 F1 键,那么将弹出 HELP 窗口显示主题 NAMEHELP 的细节内

容。而如果上面设置主题的语句换成: set topic to fldname = sys(18) .and. prgname = sys(16),则激活帮助时,将在帮助库中寻找 FLDNAME 和 PRGNAME 与此时编辑的 GET 变量名及正在运行的程序名相匹配的帮助主题,并将帮助细节显示出来。

因为可以在帮助库中添加任意多的字段,还可用任何形式的表达式来选择帮助主题,因而你可以设计出许多丰富得多的帮助机制来。

3.非标准 HELP 窗口的帮助方式

上面各种方式设计的帮助,都是将帮助信息在标准的 HELP 窗口显示的,其在多样性、灵活性等方面要受到一定的限制,如果你想设计有自己特色(譬如对颜色、窗口大小、位置等有更灵活的控制)的 HELP 方式,可参考下面仍以 USERHELP 和 USERDATA 为例的设计实例。对 HELP 实施不同的控制,将获得不同的效果。

```
C>TYPE tmp1.prg
set talk off
set proc to tmp1
select 10
use userhelp
on key = 315 do helpproc with varread(),PROGRAM()
select 1
use userdata
memovar = 0
do while .not. eof()
  @ 10,20 say "领域一的值:" get NAME
  @ 12,20 say "领域二的值:" get MONEY
  @ 14,20 say "变量三的值:" get memovar pict '99999'
  read
  skip
enddo
on key
clos data
***** 帮助子过程 *****
proc helpproc
para fieldname,programname
set compatible off
selectno = str(select(),2) &&保存当前工作区号
select 10
```

```
locate for ((fieldname = fldname).and.(programname =
prgname))
define window help from 0,0 to 10,30 panel shadow float
grow title ;
fldname+' HELP' color scheme 5 &&开自定义的 HELP
窗口
activate window help
? helptext &&在 HELP 窗口上显示帮助细节内容
wait "
select &selectno.
release window help &&关闭 HELP 窗口
return
```

本设计方法需显式地打开帮助库并进行主题定位,因而在应用程序设计时必须约定:不能用 SET HELP TO 命令隐式打开该帮助库,也不能试图去占用帮助库已占用的工作区号,否则会出错。

参考文献:

[1]严晓舟等《新一代 FOX 数据库及其实用工具》,海洋出版社

[2]《FoxBASE+ / MAC 参考指南》,国防科大 MAC 发展中心

“IBM'93 兰光系统”技术交流会

中国电子进出口总公司与美国 IBM / ETEQ 主机板在中国的独家代理——美国 MIRACLE BUSINESS INC.公司,以及北京计算机三厂于一九九三年八月二十三日在北京王府饭店召开了“IBM'93 兰光系统”技术交流会,并且在中国正式推出“兰光”系列产品。

美国 IBM 公司是世界上最大的计算机生产厂商之一,它拥有世界上最先进的生产技术。“IBM 兰光技术”是 IBM 公司的一项新的战略措施,他们力图以兰光技术更新占领原 IBM 兼容微机市场。具体而言,兰光技术就是以 120MHz 的 486 为核心的微机技术。本次活动是 IBM 公司的生产工厂第一次直接向中国用户公开“兰光技术”。他们对中国市场潜力及生产潜力,抱有强烈的兴趣,希望同国内的用户及科研单位进行技术上的合作。