

VFP 程序运行后属性值的保持

马龙军 (昆明云南财贸学院 650221)

摘要: VFP应用程序在运行过程中动态设置的属性值当应用程序运行完毕并退出后,这些动态设置的属性值就不再保存,本文给出了程序运行后保存动态设置的属性值的方法。

关键词: VFP 程序 动态设置属性值 属性值保持

1 VFP 程序运行时动态设置属性的特点

VFP是目前流行的功能强大的数据库管理系统,也是面向对象编程的常用工具之一,因而被广泛应用于管理信息系统的编程。在编程过程中,VFP可静态地在属性窗口中设置各控件的属

性,也可动态地在事件代码中设置其属性。但当程序运行结束后,VFP只保持静态设置的属性,而不能保持动态设置的属性,即不能保存在控件的事件代码中设置的属性。然而,在应用程序的编程中,经常会遇到在前一次程序运行时设置的

属性在下次运行时仍然使用,即前一次运行时动态设置的属性要保持到下次运行时仍然有效。但是,VFP并没有提供相应的功能,这给需要此功能的编程带来不便。笔者在编写一管理信息系统时,就遇到了此类问题。解决这个问题有两个途径:一是利用内存变量文件,二是动态修改控件属性。前者是每次运行表单时将属性值以内存变量文件方式保存起来,在下次运行表单时打开内存变量文件再设置保存的属性值,但采用这种方法会有以下弊端:

(1) 变量名较多时,很难记住其所代表的内容;

(2) 每次在控件的事件代码中设置属性值时,操作不统一,容易发生错误;

(3) 这样做会在磁盘上产生一个内存变量文件,增加了磁盘的负担;

(4) 一旦该内存变量丢失或复制文件时忘记了复制该内存变量文件,将无法设置保存的属性值。而对于后者,根据VFP属性的特点,能否将

表 1

字段	字段名	类型	宽度	字段说明
1	PLATFORM	字符型	8	平台
2	UNIQUEID	字符型	10	标识号
3	TIMESTAMP	数值型	10	时间片
4	CLASS	备注型	4	类
5	CLASSLOC	备注型	4	类位置
6	BASECLASS	备注型	4	基类
7	OBJNAME	备注型	4	对象名
8	PARENT	备注型	4	对象容器
9	PROPERTIES	备注型	4	属性
10	PROTECTED	备注型	4	保护成员
11	METHODS	备注型	4	方法
12	OBJCODE	备注型	4	对象代码
13	OLE	备注型	4	ActiveX 控件
14	OLE2	备注型	4	ActiveX 控件
15	RESERVED1	备注型	4	保留
16	RESERVED2	备注型	4	保留
17	RESERVED3	备注型	4	保留
18	RESERVED4	备注型	4	保留
19	RESERVED5	备注型	4	保留
20	RESERVED6	备注型	4	保留
21	RESERVED7	备注型	4	保留
22	RESERVED8	备注型	4	保留
23	USER	备注型	4	保留

表单

1 PLATFORM	WINDOWS
2 UNIQUEID	-0K2179tw2
3 TIMESTAMP	731226475
4 CLASS	textbox
5 CLASSLOC	默认
6 BASECLASS	textbox
7 OBJNAME	Text1
8 PARENT	Form1
9 PROPERTIES	Alignment = 3 Value = 120 Height = 37 Width = 133 Left = 96 Top = 48

动态设置的属性值也象静态设置属性值一样保存起来是解决这个问题的关键,经过对控件属性的分析,笔者给出了后一方案的解决办法。

2 VFP 控件属性分析

VFP编程中所使用的各控件对象的各种信息都保存在其所在的表单文件.SCX中,表单文件记录了该表单上所有对象的信息,其中包括对象名、对象所属的类与基类、对象的属性及事件代码等,该文件是一个表文件(以.DBF文件格式存储),其表结构如下:

而每一个控件对象都以一条记录的形式保存在这个表中,例如,某表单上一个textbox类控件所在记录的部分字段值是:

当运行程序时,控件的属性就是由其所在记录的PROPERTES字段的值所决定。因而,只要对PROPERTIES字段的值进行重写,就能达到动态设置属性值并保持其值的目的。

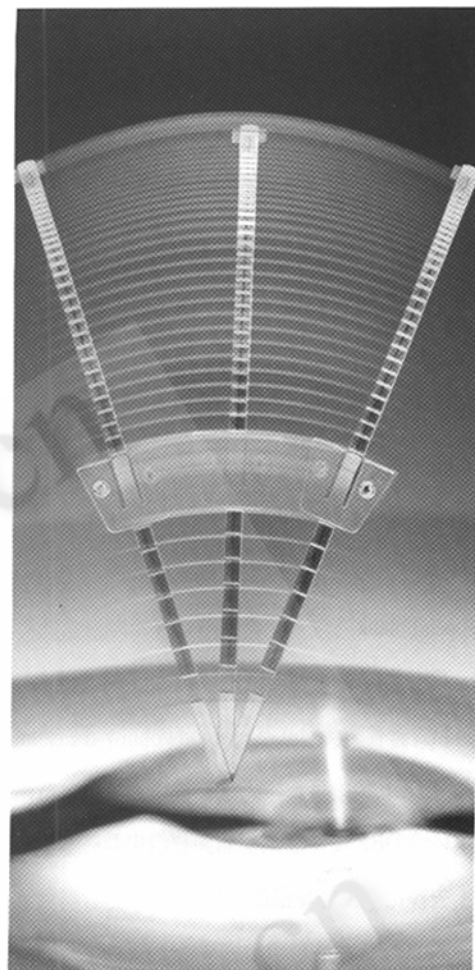
3 解决问题的具体方案

根据上面对表单文件的分析可知,只须修改控件对象的PROPERTIES字段的内容即可,由于PROPERTIES字段是备注型字段,不能象

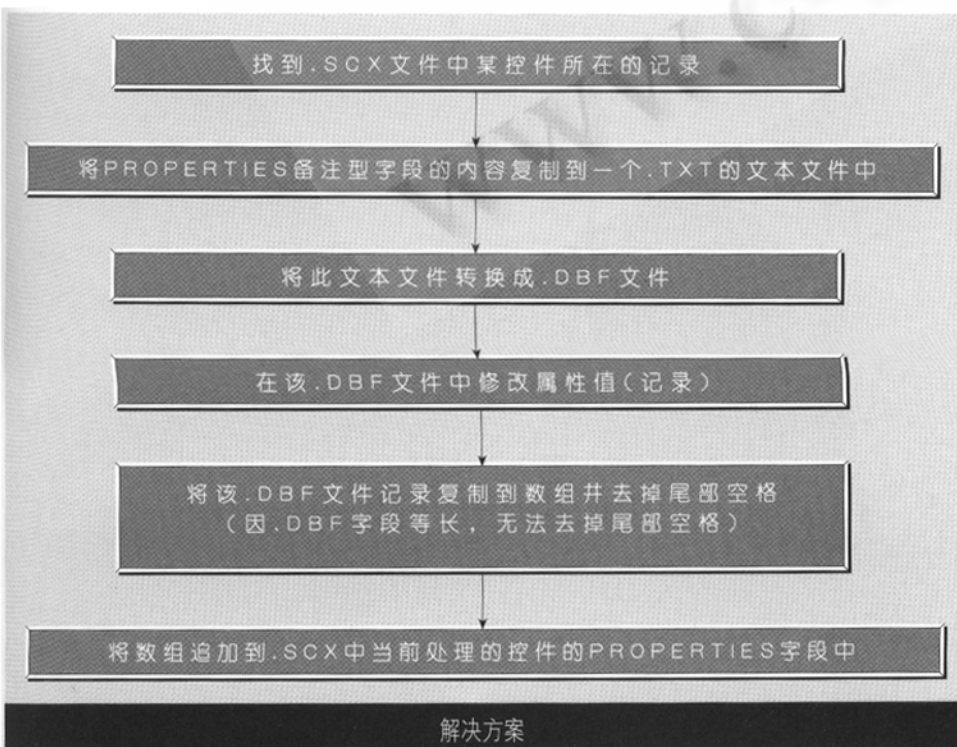
字符型字段那样直接处理,因此,只能使用对备注型字段进行操作的命令来解决这个问题,其主要解决方案是:

根据上面的思路,以表单文件ex1.scx为例,编写如下子程序(keep.prg):

```
* keep.prg
parameters cOBJNAME,cPROPNAME,
cPROPVALUE
private &cOBJNAME,&cPROPNAME
cOBJNAME=upper(left(cOBJNAME,1))+lower
(substr(cOBJNAME,2))
cPROPNAME=upper(left(cPROPNAME,1))+lower
(substr(cPROPNAME,2))
set talk off1
set safety off
set exact off
dime p-array(50,1)
* 找到某控件所在的记录
select a
use ex1.scx
loca for objname=cOBJNAME
* 将该记录的 PROPERTIES 备注字段的内容复
```



```
制成.TXT的文本文件
copy memo properties to p-txt.txt
* 将此文本文件转换成.DBF文件并删除其中的空记录
select b
create table p-dbf(属性与值 c(80))
append from p-txt.txt SDF
dele for len(属性与值)=0
pack
* 在该.DBF文件中修改属性值(记录)
do case
case type("cPROPVALUE")="N"
p= [&cPROPNAME = ] +alltrim(str
(cPROPVALUE))
case type("cPROPVALUE")="L"
if cPROPVALUE=.T.
p= [&cPROPNAME = ] + ".T."
else
```



Maintenance of attribute value after Execution of VFP program

```
p= [&cPROPNAME = ] + ".F."
endif
case type("cPROVALUE" = "C"
p= [&cPROPNAME = "cPROVALUE"]
endcase
locate for 属性与值 =cPROPNAME
if .not.eof()
replace 属性与值 with p
else
append blank
replace 属性与值 with p
endif
* 因.DBF 字段等长, 无法去掉尾部空格, 故采用将该.DBF 文件记录复制到数组中并在数组中
* 去掉每个属性尾部的空格
recnum=reccount()
copy to array p-array all
for i=1 to recnum
p-array(i,1)=trim(p-array(i,1))
next i
* 将数组追加到.SCX 中所处理的控件的 PROP-
ERTIES 字段中
select a
use ex1.scx
loca for objname=cOBJNAME
replace properties with p-array(1,1)+chr(13)
for i=2 to recnum
replace properties with p-array(i,1)+chr(13) additive
next i
* 由于只有对表单文件进行重写才能使设置的属
性值生效, 故采用向键盘缓冲区写入命令来代替
重写表单文件。
keyboard '{CTRL+PGDN}'
keyboard '{CTRL+END}'
keyboard chr(13)
```

```
keyboard '{CTRL+W}'
modify memo properties
close memo properties
close database
clear all
delete file p-txt.txt
delete file p-dbf.dbf
return
```

运行该程序时, 只需指明控件对象的名称、属性名和属性值即可。例如:

假定字符型变量 pvalue 当前的值是“第五次运行”, 要用其值修改 Label1 的 Caption 属性的值, 可用如下命令调用上面的程序:

```
do keep with "label1", "caption", pvalue
```

又如, 用数值型变量 pvalue 的值修改 Text1 的 Value 属性的值, 可用如下命令调用上面的程序:

```
do keep with "text1", "value", pvalue
```

4 结束语

在用 VFP 进行管理信息系统编程的过程中, 使用上述方法处理动态设置的属性值的保存问题是十分有效的, 利用这一方法可使表单前一次运行的结果保存到以后的运行过程中, 本程序中只给出了属性值为字符型、数值型和逻辑型且为单值的处理方法, 除此之外的属性均可参照此方法进行处理。 ■

参考文献

- 1 Visual Foxpro 及其应用系统开发, 史济民, 汤观全编著, 清华大学出版社, 2000.1.
- 2 中文 Visual FoxPro6.0 实用开发指南, 周升锋编著, 电子科技大学出版社, 1999.6.

