

在 Foxpro 程序中实现图形动态技术

刘玉 (大连电业局 116001)

宋月 (中国华录集团 116001)

摘要:本文讨论了图形动态技术的作用、实现的步骤并给出了两个实例。

关键词:图形动态技术

众所周知, Foxpro 在实现管理性数据的存储、查询及打印等方面是非常有效的。而在实际的程序开发中, 开发人员往往忽略了该系统所提供的在图形处理方面的强大功能。笔者结合在编程中的实际经验, 来简单介绍在 Foxpro 应用程序中如何实现图形的动态技术。

一、实现动态图形的用途

在实际编程中, 程序的功能固然十分重要, 如果仅仅是功能十分强大, 而界面却简单无味, 对程序的使用者来讲, 应该不能算做一个理想的应用程序。因而应用程序在实现其功能的基础上, 还应努力实现界面的友好、美观、易懂、易操作。

1. 实现动态图形, 是提供友好界面的内容之一。一个用户界面, 即使再美观、友好, 如果老是一成不变, 久而久之便会给用户一呆板、乏味的感觉, 如果让图形动起来, 可带给操作者一动感、新鲜的感觉, 使系统的界面更形象化、理想化。

2. 实现动态图形, 可使操作更形象、简单。如界面上有一计算器, 在上面击一下鼠标左键, 即将计算器激活, 拖动移至需计算的区域, 松开左键, 即可计算该区域的数据(当然计算规则要事先约定好)。再比如在操作人员不进行工作时, 系统提供一个动态的界面, 以示系统闲置; 操作人员工作时动态界面静止, 以示系统运行中。

二、实现动态图形的设计原理与步骤

在 Foxpro 应用程序中, 实现动态图形的原理是首先设计几个不同状态的图形(可在画笔里或其他工具里绘制), 然后按一定的轨迹(直线、弧线或其他轨迹)变换这些图形, 利用人眼的视觉暂留, 便会形成图形运动起来了的效果。

根据上述设计原理, 可得实现动态图形的设计步骤如下:

1. 根据实际的需要, 设计几个不同状态的图形(这需要有美术的基本知识);
2. 确定图形的运行轨迹, 以及特殊点上的图形状态(这须具备一定的数学知识和逻辑知识);
3. 将不同状态的图形嵌入运行轨迹中, 并通过编程实现。

三、实现动态图形的事例研究

下面通过两个具体事例, 来简单分析如何实现图形的动态。

事例 1: 实现在一定区域内, 有一只活动的小猫在往复运动。

首先需要设计几只处于不同状态的小猫, 笔者在画笔里简单地画了以下几只小猫。



图 1 各种状态的小猫

然后确定小猫的运行轨迹为直线往复运动, 以及直线两个端点的位置坐标, 运动的步长。

最后将各种状态的小猫嵌入运行轨迹中。即在直线的两个端点为状态 3, 在直线中为状态 1 和 2(或 3 和 4) 互相交替出现。

具体程序如下:

```
COL = 70
QFH = ' - '
DO WHIL .T.
```

```

WW = INKE(0.1)
IF WW = 27
EXIT
ENDI
@20, COL CLEA TO 23, COL + 10.5 COLO RGB(192,
192, 192, 192, 192)
IF COL < 70 AND QFH = ' + '
COL = COL + 2
ELSE
IF COL = 70 OR COL = 12
QFH = IIF(QFH = ' - ', ' + ', ' - ')
IF COL = 70
COL = COL - 2
ENDI
ENDI
IF COL > 12 AND QFH = ' - '
COL = COL - 2
ENDI
ENDI
DO CASE
CASE COL = 70 OR COL = 12
@ 20, COL SAY (LOCFILE("cat3. bmp", "BMP | ICO",
"Where is cat3?")) BITMAP ;
SIZE 2.846, 10.400 ;
STYLE "T"
CASE COL < 70 AND QFH = ' + '
IF MOD(COL/2, 2) > 0
@ 20, COL SAY (LOCFILE("cat4. bmp", "BMP |
ICO", "Where is cat3?"))
BITMAP ;
SIZE 2.846, 10.400 ;
STYLE "T"
ELSE
@ 20, COL SAY (LOCFILE("cat5. bmp", "BMP |
ICO", "Where is cat3?"))
BITMAP ;
SIZE 2.846, 10.400 ;
STYLE "T"
ENDI
CASE COL < 70 AND QFH = ' - '
IF MOD(COL/2, 2) > 0

```

```

@ 20, COL SAY (LOCFILE("cat1. bmp", "BMP |
ICO", "Where is cat3?"))
BITMAP ;
SIZE 2.846, 10.400 ;
STYLE "T"
ELSE
@ 20, COL SAY (LOCFILE("cat2. bmp", "BMP |
ICO", "Where is cat3?"))
BITMAP ;
SIZE 2.846, 10.400 ;
STYLE "T"
ENDI
ENDC
ENDD
事例 2: 实现动态图形拖动技术。要求: 在某区域有一图形, 在此区域按下鼠标左键, 拖动图形至某一区域, 松开左键, 完成一相应操作。如下程序拖动一计算器至需计算的区域, 松开左键, 按事先约定的规则进行数据计算。
SAVE SCRE TO XX
DO WHIL .T.
REST SCRE FROM XX
@ MROW(), MCOL() SAY (LOCFILE("jsq. bmp", "
BMP | ICO", "Where is jsq?"))
BITMAP ;
SIZE 2.68, 7.000 ;
STYLE "T"
EXX = MROW()
EYY = MCOL()
SSWW = INKE(0.1, 'HM')
@ MROW(), MCOL() SAY (LOCFILE("jsq. bmp", "
BMP | ICO", "Where is jsq?"))
BITMAP ;
SIZE 2.68, 7.000 ;
STYLE "T"
IF SSWW = 151
&& 此处删掉了数据计算内容
EXIT
ENDI
ENDD

```

(来稿时间: 1997年6月)