

VFP 下组合查询类和通用查询类的设计及应用

王玲 顾一禾 王平立 (南京理工大学计算机系 210094)

摘要:本文介绍了 Visual FoxPro 3.0 环境下组合查询类和通用查询类的设计实现方法,这些类对任何 MIS 系统都是适用的,并且不需要进行任何修改。文中还给出了组合查询类和通用查询类的详细设计思路和关键的原程序清单,并有具体的应用实例。

关键词:类 继承 表单 基类 组合查询

在研制 MIS 系统的过程中,组合查询是我们经常要使用的功能,组合查询的设计要求是允许用户选择一个或多个数据库字段组成复合条件以便进行记录的快速定位。我们注意到,各种各样的组合查询界面都是大同小异,完全可以进行统一的设计,既节省了时间,又保证了程序设计风格的一致。Visual FoxPro 3.0 为解决这一问题提供了丰富的工具。

Microsoft Visual FoxPro 3.0(以下简称 VFP 3.0)是一个 32 位的面向对象的数据库开发系统,运行于 Windows 3.1(具有 Win32s)、Windows 95 和 Windows NT 操作系统,可借助于工具条、对象、可视控制来自动完成界面的设计并执行各种任务,不仅如此,VFP 3.0 中还有各种各样的基类,用户可以重复使用这些类,直观、创造性地建立应用程序。

在开发 MIS 系统的过程中,充分利用 VFP 3.0 提供的面向对象的开发环境,特别是关于类的设计思想,将组合查询功能以及其他通用录入、查询浏览功能都定义成类,封装在自己的类库 MYLIBS.VCX 中,在以后的程序设计中可重复使用类库中的类,减轻了程序设计的负担,保证了程序的质量。

下面将利用 VFP 3.0 组合查询类的方法介绍给大家。

一、VFP 3.0 中类的特征和定义方法

类的特征主要有三个:即封装性、子类特性、继承性。

·封装性:一个对象的所有性质、事件和方法全部在类定义中说明;

·子类特性:一个已存在类的子类可以拥有其父类的全部功能,在此基础上,可添加其他控制和功能;

·继承性:如果你改变一个类,这个变化将反映在基于这个类的所有子类中;

类的定义也有三种方法:

1. 用程序设计的方法定义类,即使用 CREATE CLASS 命令;
2. 可视化定义类;
3. 将 FORM(即表单)作为类保存;

二、组合查询类的设计及实现

我们是利用第二种方法设计组合查询类。

打开 VFP3.0 的类设计器,基于 VFP3.0 提供的 FORM 类构造组合查询类,类的名称为 Searchform1,并存放在类库 MYLIBS.VCX 中,如图 1 所示:

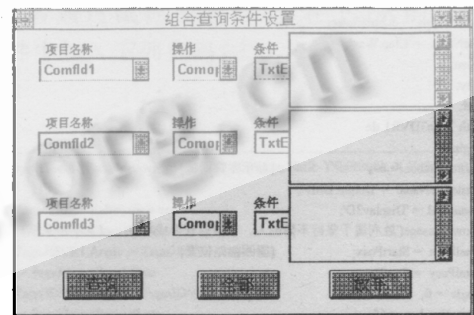


图 1 组合查询条件设置

在图 1 中,我们设置了三项条件,也就是说可以任意三项条件进行“与”组合。

为了保证设计的组合查询类对任意应用系统通用,我们的设计步骤如下:

1. 将 Searchform1 的 Caption 改为“组合查询条件”
2. 将 Searchform1 的 BackColor 属性改为 0, 255, 255, 即背景为浅蓝色;
3. 在 Searchform1 中加入六个下拉列表框基类 Com-

boBox, 将项目名称下的三个下拉列表框的 Name 属性分别改为 Comfld1, comfld2, comfld3, 而操作下的三个下拉列表框的 Name 属性分别改为 Comop1, comop2, comop3, 项目名称下的三个下拉列表框 Comfld1, comfld2, comfld3 用来列出查询数据库中所有除类型为“G”、“M”、“U”的字段, 而操作下的三个下拉列表框列出所有的条件操作符。

Comfld1, comfld2, comfld3 和 Compop1, Compop2, Compop3 的初始化程序(即它们的 INIT 事件)因篇幅关系略。

4. 在 Searchform1 中加入三个文本框基类 TextBox, 将 Name 属性分别改为 Txtexpr1, Txtexpr2, Txtexpr3, 用来录入查询条件;

5. 在 Searchform1 中加入三个列表框基类 ListBox, 将 Name 属性分别改为 Lstexpr1, Lstexpr2, Lstexpr3, 当某字段的值可以从一个表中选择时使用列表框, 否则使用文本框。也就是说, 当我们在项目中选择一个字段时, 如果该字段的 Comment 含有一个表的名字, 则条件的输入选择列表框, 这时操作符固定为“在内”, 用户可以从列表框选择一个或多个条件值, 并将选择的条件值送入相应的文本框中。Comfld1, comfld2, comfld3 的 Interactivechange 事件根据选择的字段进行文本框和列表框的转换, 而 Lstexpr1, Lstexpr2, Lstexpr3 的 Lostfocus 事件则将列表框中选择的条件值进行组合并分别送入 Txtexpr1, Txtexpr2, Txtexpr3 中, 程序清单略。

6. 在 Searchform1 中加入命令组控件 CommandGroup, 将 Name 属性改为 CommandGroup1, ButtonCount 属性改为 3, 也就是说, 命令组中有三个命令, Command1 的 Caption 属性改为“查询”, Command2 的 Caption 属性改为“全部”, Command3 的 Caption 属性改为“退出”, 当我们用鼠标按下任一命令按钮时, 将执行下面的程序。

Commandgroup1 的 Click 事件程序设计清单:

```
# DEFINE NORECSFOUND-LOC "没有查到符合条件的记录, 请再输条件!"
```

```
# DEFINE ALLRECSFOUND-LOC "全部记录数为:"
```

```
# DEFINE SEARCHFOUND-LOC "符合条件的记录数为:"
```

```
DO CASE
```

```
  CASE THIS.VALUE = 1
```

```
    LOCAL cGetExpr
```

```
SET FILTER TO
```

```
m. cGetExpr = THISFORM. SearchExpr()
```

```
IF ! EMPTY(m. cGetExpr)
```

```
  SET FILTER TO &cGetExpr
```

```
  LOCATE
```

```
  * Check if no records found
```

```
  IF EOF()
```

```
    = MESSAGEBOX(NORECSFOUND-LOC)
```

```
    THIS.Parent. ChangedFilter = .T.
```

```
  RETURN
```

```
ENDIF
```

```
THIS.Parent. ChangedFilter = .F.
```

```
COUNT TO m. ncount
```

```
= MESSAGEBOX(SEARCHFOUND-LOC + STR(m. ncount, 4))
```

```
GO TOP
```

```
ELSE
```

```
  LOCATE
```

```
ENDIF
```

```
CASE THIS.VALUE = 2
```

```
  SET FILTER TO
```

```
  THIS.Parent. ChangedFilter = .F.
```

```
  COUNT TO m. ncount
```

```
= MESSAGEBOX(ALLRECSFOUND-LOC + STR(m. ncount, 4))
```

```
GO TOP
```

```
OTHERWISE
```

```
ENDCASE
```

```
RELEASE THISFORM
```

“查询”命令按钮完成的功能是根据查询条件过滤当前表, 如果有符合条件的记录, 则显示符合条件的记录个数, 并且记录指针指向第一个符合条件的记录; 如果没有, 则提示“没有查到符合条件的记录, 请再输条件!”。“全部”命令按钮完成的功能是取消过滤条件, 将表恢复成全部记录, 并将指针定位在第一个记录上。

以上就是组合查询类设计及实现的大致过程, 下面谈谈如何使用组合查询类。

三、组合查询类 Searchform1 的使用

我们设计的组合查询类是设置查询条件并显示符合条件的记录数, 而没有将记录的具体内容显示在屏幕上,

因此我们基于 VFP3.0 提供的 Container 类设计了通用查询类 Search1, 并存放在类库 MYLIBS.VCX 中, 如图 2 所示:

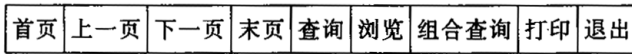


图 2 通用查询类 Search1

在 Search1 包含类中, 我们将 VFP 3.0 提供的 WIZ-STYLE.VCX 类库的 Txtbtms 类加入到 Search1 中, 因此 Search1 包含类继承了 Txtbtms 的所有特性, 在图 2 中, 我们设计的 Search1 类由九个命令按钮组成, 它们的 Caption 分别是“首页”, “上一页”, “下一页”, “末页”, “查找”, “浏览”, “组合查询”, “打印”, “退出”等, 在这个类中, 可使用我们上面建立的组合查询类, 使用方法是在组合查询按钮的 click 事件中加入下面一段程序:

```
LOCAL oSearchDlog
oSearchDlog = CREATE("searchform1")
oSearchDlog.SHOW()
THIS.Parent.TopFile = .F.
THIS.Parent.EndFile = .F.
THIS.Parent.NavRefresh()
```

其中, NavRefresh() 是因为 Search1 继承了 Txtbtms 类的事件。关于其他按钮的设计这里不再讨论。

四、通用查询类 Search1 的使用

通用查询类 Search1 可以不加修改地应用于任何系统中, 下面我们以一个具体的实例来说明。在人事管理系统中, 我们在数据库中定义人员信息表(RYXX.DBF), 对人员信息的查询我们是通过一个表单进行, 将表单的 Caption 属性改为“人员基本信息查询”, 再根据查询的需要, 在表单中加入各种控件以显示表中的内容, 并且加入通用查询类 Search1, 下图是进入人员信息查询后用鼠标按一下“组合查询”按钮的显示结果, 此时用户可以根据屏幕提示输入组合查询条件。

下图中, 当前表 RYXX.DBF 中除类型为“G”、“M”、“U”的所有字段将加入到项目下的下拉列表框中, 而所有的操作符则加入到操作下的下拉列表框中, 用户可以选择一个或多个表中的字段组成复合条件进行记录的快速定位, 目前我们的条件有三项, 也就是说可以任意三项条件进行“与”组合。在设置一个条件时, 用户首先在项

目名称下的下拉列表框中选择一个项目, 比如职工号、文化程度、性别等, 我们有两种类型的项目, 一种是“直接输入条件”项目, 在定义该项目的字段时, 其 Comment 为空。比如说 RYXX.DBF 中的职工号, 用户在选择“操作(等于、不等于、小于、大于等)”之后, 直接输入条件; 另一种是“选择条件项目”, 这时“操作”栏被屏蔽掉, 其操作符固定为“在内”, 而条件处是一个列表框, 列表框的内容为一个表的全部记录, 表的名称在字段的 Comment 中规定, 用户可选择其中一项或多项作为输入, 这样就设置好了一个条件, 另两个条件的设置与此类似。条件设置完毕后, 选择“查询”按钮, 系统将提示符合条件的记录数有多少。此时 Search1 类中的“首页”, “上一页”, “下一页”, “末页”, “浏览”等命令显示的都是符合条件的记录。

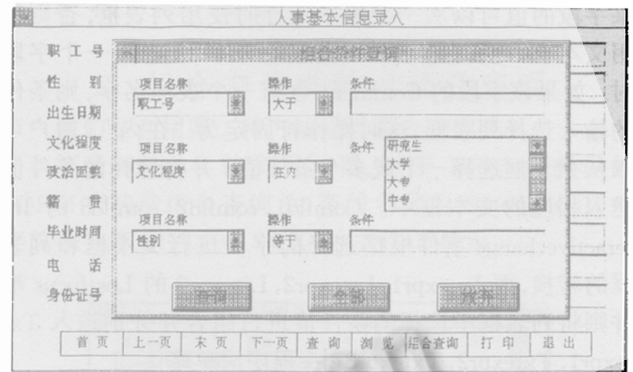


图 3 通用查询类 Search1 的应用

本文介绍的组合查询类 Searchform1 和通用查询类 Search1 可以应用在任何 MIS 系统的查询程序中, 并在 Windows 3.1、Windows 95、Visual FoxPro 3.0 环境下运行通过。

参考文献

- [1] 运通创作室编著《Visual Foxpro 3.0 语言实用详解》学苑出版社
- [2] 彭祺, 曹晓阳编《Visual Foxpro 3.0 面向对象编程》成都科技大学出版社

(来稿时间: 1997年11月)