

# 信息管理系统中开放式数据库的建立及维护

山东工业大学数理系 齐元华 周灿林  
 山东工业大学电力学院 谷廷坤  
 山东中创软件公司 张清



**本文介绍 VFP 下信息管理系统设计中用户自定义项目的数据库录入、查询及打印输出的实现。**

任何信息管理系统都必须具有三个基本的模块：数据录入、数据查询和数据打印输出。一般情况下，这些处理都是对程序员设计好、固定不变的数据库、表进行的。系统设计过程中，程序员必须了解信息管理工作的具体内容，以决定需要建立哪些数据库、表。这些数据库、表一旦建立，其结构对用户来说是固定的，使用过程中不能修改，数据查询内容和打印输出报表也是对结构不变的数据库、表设计的。这样的系统只能处理预先设计好的项目。在信息管理实际工作中经常会出现一些新的问题，需要形成项目。这就要求信息管理系统数据库应具有开放性的特点，允许用户修改其结构。但如前所述，目前多数系统的数据库都是封闭式的。为此，笔者在“山东省派出所档案管理系统”的设计中尝试增加这种功能，在常规管理项目之外，让用户能够根据实际工作的需要自由增加新项目。对新增项目，用户可以象常规项目一样对它们进行数据录入、数据查询、数据打印输出等处理。系统在山东省 1000 多个派出所使用以来，此功能深受用户的好评。本文将介绍这项功能的设计与实现过程。上述系统用 VFP5.0 编写，其他数据库软件也可采取类似的方法。新项目的处理共分四个方面，以下分别介绍。

## 填加项目

填加项目是在特定的数据库中增加一个数据表。首先根据用户输入的信息建立一个自由表，然后将该表填加到数据库，并在数据库中保存每个表字段的标题信息。用户输入的信息包括新增项目的名称、对应数据表各字段的

标题(Caption)、字段类型、字段长度等。填加项目表单布局下。其中“下一个字段”命令按钮 Click()事件的代码将用户输入的每个字段的类型、中文名(Caption)、字段宽度保存到三维数组 thisform.zdxx [N,3] 中去，“确认填加”命令按钮的 Click()代码按照数组 thisform.zdxx [N,3] 的内容创建新表并增加到数据库中，同时项目的名称和自动产生的表文件名被存储到一个自由表中以便项目删除模块使用。本例中该自由表名 nbglxm，数据库名为 nbgl，记录细目信息的数据表名为 xmX.db (x 为一数字，表示该项目的序号)。

“确认填加”命令按钮的 Click()代码如下：

```
oldalias=alias()
nbstr=currentpath+ "\ data \ nbglxm"
use &nbstr in 0
goto bottkk=val(right(alltrim(kwj),1))+1
use
biao=currentpath+ "\ data \ xm" +alltrim(str(kk))
xm= "xm" +alltrim(str(kk))
biaot=biao+ ".dbf"
if file(biaot)
delete file &biaot
endif
sjlx=thisform.zdxx [1,1]
if sjlx== "D"
create table &biao (zd1 &sjlx)
else
create table &biao (zd1 &sjlx (thisform.zdxx [1,3]))
endif
for i=2 to thisform.zdn
```

```

sjlx=thisform.zdxx [i,1]
zdm="zd"+alltrim(str(i))
if sjlx=="D"
alter table &xm add column &zdm &sjlx
else
alter table &xm add column &zdm &sjlx (thisform.
zdxx [i,3])
endif
next
for i=1 to thisform.zdn
zdm=xm+ ".zd" +alltrim(str(i))
=dbsetprop (zdm, 'Field', 'Caption', thisform. zdxx
[i,2])
next
use
select 0
nbstr=currentpath+ "\ data \ nbglxm"
use &nbstr exclusive
append blank
replace xmmc with thisform.text1.value
replace kwjm with "xm" +alltrim(str(kk))
use
select &oldalias
thisform.release
    
```

### 数据查询

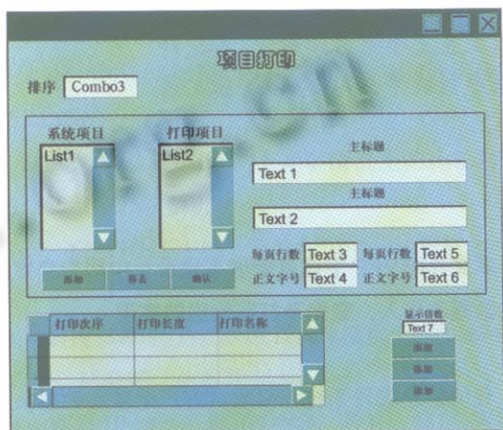
查询工作首先调用dbgetprop()函数读取新增项目各字段变量名及标题名，并将它们存储到二维数组thisform.mitem中去，数组thisform.mitem的标题列被用作下拉式列表框combol的rowsource，combol用于选择查询字段。这样选择某字段后，通过与数据库对应表的标题比较，找到字段变量名，根据输入的数值形成查询表达式。新增项目各字段变量名及标题名的获得在查询表单的init()过程中完成。

### 数据打印

对于用户自己填加的打印项目，因为程序员事先并不知道其字段的个数、每个字段的数据类型、字段宽度、标题等信息，因而无法设计报表进行数据输出。这种情况下，采取自由打印方式较为方便。

自由打印的输出方式分两步实现，首先读取某一项目所有字段的标题信息，将其置于一数组中，并用数组作

为一下拉式列表框的数据源，这样通过鼠标的单击操作，用户可以根据不同的需要灵活选择欲输出的字段。接着程序读取用户所选字段的字段名称、字段长度信息填写自由打印数据表。并将此表作为一表格控件的数据源，通过表格控件，用户可以设置某字段的打印长度，各字段的打印顺序等。当某字段的打印长度小于字段长度时，该字段将折行打印。自由打印字段设置表单如图所示。



### 项目删除

随着时间的推移有些管理项目将变的过期，应适时的加以删除。通过项目列表框，用户选择欲删除项目。确认删除按钮Click()事件将记录选中项目的数据表从数据库中移出并删除，同时，记录项目名称和数据表名称的自由表中有关该项目的记录也被删除。Click()事件的代码如下：

```

nbstr=currentpath+" \ data \ nbglxm"
use &nbstr exclusive
n=reccount()
skip 3
for j=1 to thisform.list2.listcount
for i=1 to n-3
if thisform.list2.list [j] ==alltrim(xmmc)
ll=alltrim(kwjm)
remove table &ll delete
delete
exit
endif
skip
next
next
pack
use
    
```