

使用 Visual FoxPro 创建自动服务程序

南京师范大学地理科学院遥感所 韦玉春
陕西省教育学院图书馆 刘 莉

建立自动化服务程序，对于程序的重用和系统集成具有重要的意义。本文集中讨论了使用 Visual FoxPro 建立自动服务程序应注意的问题，并给出了一个应用实例。

自动服务程序是一个组件应用程序，其他应用程序通过自动服务程序完成自己的任务。现在，Visual FoxPro(VFP) 提供了经过改进而更可靠的自动服务程序，能更好地与 Microsoft Transaction Server、Microsoft Visual Basic(VB) 和 Active Server Pages 等产品协调工作。

使用 VFP 创建自动服务程序，可以用来显示可重复使用的表单、或将一个复杂例程打包进一个简单组件中供其他程序员使用。另外，还可以创建一个或多个处理企业级商务规则的服务程序，供使用该规则的其他客户应用程序使用，以保证企业内部数据的一致性。

本文以 VFP 建立自动服务程序，以 VB 建立客户程序为

例对此进行说明，语言的版本均为 6.0，工作平台为 Windows NT4.0。

建立自动服务程序应注意的问题

1. 自动服务程序类型

在 VFP 中，可以创建一个进程外或进程内的自动服务程序。进程外服务程序是一个有独立进程的可执行(.exe) 文件，可以包括用户的界面，并且可以单独运行。进程内服务程序是一个动态链接库(.dll) 文件，与调用的客户程序运行在相同的进程地址空间，不能包括用户的界面，否则将产生错误。

另外, 进程外 .exe 服务程序可以用 VFP6.0 新添加的函数 SYS(2335)进行设置, 例如关闭远程执行的事件。这些事件由用户定义的模式表单、系统对话框、MESSAGEBOX() 函数和 WAI 令等创建。

2. 异常处理

如果发生异常, VFP 自动服务程序可以平稳地中止服务, 此时, 自动服务程序将设置 COM ErrorInfo 对象 (通过 IErrorInfo), 并取消当前的方法程序。客户程序通过访问 COM ErrorInfo 对象中的信息, 可以选择是释放 VFP 自动服务程序, 还是对异常情况进行处理。VFP 中有一个函数 COMRETURNERROR(), 用来处理自动服务程序中发生的错误。COMRETURNERROR() 用在 Error 方法中, 向 COM 中加入信息, 供客户程序确定自动服务程序中错误的来源。详细内容, 请参阅语言参考手册。

使用 VFP 的函数 COMARRAY(), 可以指定向 COM 对象传递数组的方式, 并且可以指定该数组是基于 0 的还是基于 1 的 (只有向 COM 对象传递数组时, COMARRAY() 函数才有实际的意义)。

3. 从项目中连编 .dll 和 .exe 文件

由于进程内 .dll 和进程外 .exe 自动服务程序是通过类的实例来激活的, 因此没有必要指定主文件。但是, 如果想单独地使用 .exe 自动服务程序, 就必须建立并指定主文件。

4. 项目名和类型库的说明

项目名和类型库的说明在“项目信息-服务程序”页。该页的信息只有在进行第一次连编后才会出现。换句话说, 当前的修改只有连编后才能在该页反映出来。我们不建议修改该页码中的项目名称, 因为该名称与连编组件的注册信息有关。还有, 项目名和类型库说明不要使用汉字, 否则其他的程序将难以调用建立的类库 (汉字信息有时在对象浏览器中不可识别), 使用英文或拼音总是比较好的选择。

5. 用户界面中的注意事项

(1) 关闭表单时不要使用 release 方法, 因为 release 是与 do form 语句对应的, 不适合组件的情况。要使用 hide 方法或其他的方式。

(2) 不要使用表单的数据环境。所有的表, 应该在表单的 load 事件中打开, 并建立关系。不要放在 init() 事件中, 因为表单的 init() 事件不仅发生在 load () 后, 而且发生在表单内所有控件的 init() 后, 而表单中的控件可能

使用有关的表。

(3) 所有表单中要使用的公共程序和函数, 应该编入到单独的用户类中。

在用户类中, 建议建立一个 zSetEnv 方法, 用来初始化数据库的基本环境, 并引入基本的参数。该类作为一个父类, 应该在每个表单中使用。

对于原有应用系统中使用的全局变量, 可以将其改变为属性, 然后, 在对象建立之后、显示之前给该属性赋值:

```
O1=createobject('myobj')
O1.myprop=new_value
O1.show
```

对于服务程序和客户程序都要使用的参数, 应该保存在独立的配置文件中。

(4) 表单的 showWindow 应该根据调用方式进行设置。

如果自动服务程序为外部程序例如 VB 调用, 表单的 showWindow 应该设置为“顶层表单”, 如果为 VFP 内部调用, 则可以使用“在屏幕中”。

建立一个进程外的服务程序实例

由于能够带有用户界面和独立运行两个特点, 建立进程外的自动服务程序更具有重要的意义。总体上讲, 建立自动服务程序包括 8 个步骤: 建立自动服务程序类, 进行测试, 设置相关的参数, 注册, 进行调用测试, 系统打包, 安装和总体测试。

1. 在 VFP 内创建自动服务程序类

在 VFP 中创建一个自动服务程序, 只需要在项目中包含定义为 OLEPUBLIC 的类。项目中可以包含任意多个 OLEPUBLIC 类, 各个类可以在程序 (.prg) 中或类库 (.vcx) 中定义。

下面, 我们通过建立一个进程外组件 VCOM 进行说明。

在 VFP 内, 建立一个项目, 名称为 VCOM。然后, 在类的设计器中建立我们需要的类。下面是类浏览器中显示的一个类库 vfpcomx.vcx, 其中带有 * 号的是 OLEPUBLIC。

类库中的 appx 是公共的函数库, 包括有多个属性和方法。下面给出了其中的一个属性 cWrkPath (工作的目录) 和方法 zsetenv (用来设置工作环境, 打开相应的数据库), 基本的代码为:

* 程序中, d:\docmag 为项目的目录

```

*-- 类: appx (d:\docmag\libs\vfpcmx.vcx)
*-- 父类: custom
*-- 基类: custom
#include "d:\docmag\libs\vfpcmi.h"
DEFINE CLASS appx AS custom OLEPUBLIC
    cwrkpath = 'd:\docmag\data'
    Name = "appx"
    PROCEDURE zsetenv
        parameters cWrk
        if !empty(cWrk
            this.cWrkPath=cWrk
        else
            this.cWrkPath='d:\docmag\data'
        endif
    SET DOHISTORY OFF
    SET TALK OFF
    SET CONSOLE OFF
    SET MULTILOCKS ON
    SET SYSFORMATS ON
    SET CONFIRM ON
    SET CENTURY ON
    SET DATE TO YMD
    SET HOURS TO 24
    open database this.cWrkpath+' \wycdoc'
    set database to wycdoc
    set default to (this.cWrkPath)
ENDPROC
* 其他方法和事件 ...
*
PROCEDURE Destroy
    close tables all
    close database all
ENDPROC
ENDDEFINE

```

您可以使用类设计器方便地建立不同的类。建立完后，在菜单“类-类信息”对话框中选择“OLE 公共”，就可以指明该类为 OLEPUBLIC 类。

上面的类库中，fmdataType 和 DocQuery 是用户界面，用来进行数据的输入和查询。在各个表单中，均继承了父类 APPX(例如，上图中的 fmdatatype 中的 appx0)。

FmdaType 类的关键代码如下所示：

```

*-- 类: fmdatatype (d:\docmag\libs\vfpcmx.vcx)
*-- 父类: form
*-- 基类: form
DEFINE CLASS fmdatatype AS form OLEPUBLIC
    * 主要的属性设置，ShowWindow 必须依据应用的
    需要设置
    ShowWindow = 2
    AutoCenter = .T.
    Caption = "编辑档案"
    * 各个按钮的属性 ....
    * 按钮中相关的代码，zOptab 和 zSetBuffer 是在
    Appx 定义的方法
    PROCEDURE Load
        with this.aPPx0
            .zOptab('doctype')
            .zsetBuffer('doctype',5)
        endwhile
    ENDPROC
    * 其他方法 ...
ENDDEFINE

```

2. 测试

在 VFP 的命令窗口，使用下面的代码进行测试：

```

public oMain,oForm
oMain=createObject('appx')
oMain.zsetenv('d:\docmag\data')
oForm=createobject('fvfpm')&& 内部的类，系统的
主界面
oForm.show

```

测试通过后，建立主文件，用来单独运行该组件。其程序是在上面代码的前面加上类库的路径，例如：

```

set classlib to d:\docmag\libs\vfpcmx addi
在上面代码的后面加上消息循环：
read events

```

测试通过后，进行连编。一定要注意，因为包括了界面，所以必须编译为 EXE 文件。

3. 设置相关参数

通过“项目-项目信息-服务程序”页进行设置。系统默认实例为“多重使用”，应该根据需要确定该参数。要注意的是，如果修改了“服务程序”选项卡中的内容，为了使新的设置生效，需要重新连编 .dll 或 .exe。一

般的情况下，不要进行修改。

4. 注册自动服务程序

连编自动服务程序时，系统自动在当前的机器上注册。使用 Visual FoxPro“安装向导”创建安装盘时，安装程序会在用户的机器上注册。如果要人工注册服务程序，可以参照如下进行（假定 VCOM 是我们的服务程序名）：

对于进程外：VCOM /regserver

对于进程内：REGSVR32 VCOM.dll

如果要取消注册，相应的为：VCOM /unregserver
和 REGSVR32 /u VCOM.dll

5. 组件文件说明

连编项目后，系统创建了三个文件。

(1) .dll 或 .exe 文件。本例建立的是 VCOM.exe

(2) 一个类库 (.tlb) 文件，例如 VCOM.tlb。这是一个二进制文件，列出了自动服务程序上所有发布的类以及它们的属性、方法和事件。OLE 对象浏览器可以读到这些信息，并显示它们。

(3) 一个注册 (.vbr) 文件，例如 VCOM.vbr。注册文件列出了您的服务程序中类的全局唯一标识符(global unique Ids) (GUID)，这是一个文本文件。注意其中的 PROGID 项，它是调用对象时要使用的参数。

例如：

```
... \PROGID=vcom.appx
```

意味着在程序中调用方式为(VB)：

```
OO=createobject("vcom.appx")
```

其中，OO 为程序中要使用的对象

6. 调用测试

使用 VB 系统，对组件进行测试。

首先建立一个标准类型的项目，加上一个窗体，例如 form1。然后通过菜单“工程-引用”

选取要引用的组件，组件的说明即我们在 VFP“项目信息-服务程序”中见到的类库说明。按 F2 打开对象浏览器，选择 VCOM 库，就可以看到我们建立的类了。接下来需要定义几个全程

对象变量，建立对象的引用。代码如下：

```
Dim oAV As Object, oAPPo As Object
Dim cPath As String
Private Sub cmCall_Click()
oAV.Show
End Sub
Private Sub Form_Load()
```

```
cPath = "d:\docmag\data"
```

```
set oAPPo = CreateObject("vcom.appx")
```

```
oAPPo.zsetenv (cPath)
```

```
Set oAV = CreateObject("vcom.fmdatype")
```

```
End Sub
```

编译该程序为 EXE 文件，运行该 EXE 文件进行测试。为安全起见，我们不建议在 VB 的设计环境下进行测试。

7. 系统打包

利用各语言提供的打包程序建立可分发的文件。用 VB 打包时要注意将有关组件的注册文件包括在内（例如 VCOM.vbr）。使用 VFP 打包时，要注意“步骤 2-指定组件”，在这里：

(1) “VFP 运行时刻库组件”是必选的

(2) 要选中“COM 组件”，加入您编写的组件，例如 VCOM.exe，并选中“安装远程 COM 组件”

(3) 根据您的组件是否调用了其他的控件等，选择其他的选项。

如果这一步不选择“COM 组件”，而在第 6 步将它作为“ActiveX”，安装程序将失败。至少我的测试是这样的结果。

8. 安装测试

找一台没有 VB 和 VFP 的微机，进行安装测试。其顺序是先安装组件，然后安装调用组件的程序。安装结束后，运行客户程序。如果通过了测试，那么，说明您可以进行组件的作业了。使用自动服务程序，尤其适合于进行协作式开发。项目组中的程序员可以使用自己擅长的语言编写程序，然后通过对象的引用建立系统。在当前计算机的速度不断提高，内存不断增加的情况下，自动服务程序具有明显的便利性和可维护性，是值得大力提倡的。■

参考文献

- 1 Microsoft Corporation 著，Visual FoxPro 6.0 中文版程序员指南。北京希望电脑公司，1998
- 2 陈羽，麦中凡，从 OLE 到 DCOM-微软的构件软件技术分析。中国计算机用户，5,1997
- 3 范伟宏，赵征西，龚育昌，构件对象模型。计算机科学，5，1997
- 4 米里亚姆·利斯金编著，Visual FoxPro 专家方案，上海远东出版社，199