

文章编号:1671-6833(2004)01-0070-03

在 Visual Basic 中嵌入 Delphi 窗体的应用技术

周振红¹, 郭恒亮¹, 杨国录²

(1. 郑州大学环境与水利学院, 河南 郑州 450002; 2. 武汉大学水利水电学院, 湖北 武汉 430072)

摘要: Delphi 窗体嵌入到 Visual Basic 应用中, 可充分利用 Delphi 的可视化窗体继承技术, 并为项目组在语言工具选择上提供灵活性. 在 Delphi 的 DLL 项目中添加窗体, 并在其导出过程中完成窗体的创建、显示和释放; Visual Basic 应用通过调用 DLL 的导出过程即可展示 Delphi 窗体. 实现的关键是在导出过程的参数表中设置句柄参数, 以便传递调用程序对象的实例句柄; 若显示非模式窗体, 导出过程还需返回窗体句柄, 以便调用程序对窗体进行维护.

关键词: Delphi; 可视化窗体继承; Visual Basic; DLL

中图分类号: TP 311 **文献标识码:** A

0 引言

窗体设计器能极大地提高开发应用程序的整体效率, 因此 Delphi、Visual Basic 等 RAD 工具都拥有窗体设计器. Delphi^[1] 窗体设计器的独特之处在于: Delphi 是建立在一个真正面向对象的框架结构基础之上的, 它的可视化窗体继承(Visual Form Inheritance; VFI) 技术允许动态继承窗体, 从而达到最大程度重用窗体对象及其实现代码的目的.

Visual Basic^[2] 在快速原型、系统集成方面仍有其不可替代的作用. 将 Delphi 窗体嵌入到 Visual Basic 环境中, 一方面可利用 Visual Basic 开发应用的简便高效和 Delphi 可视化窗体继承技术的优势; 另一方面也为项目组在语言工具的选择上提供宽松的环境.

开发基于 Windows 平台的客户端应用, 且与语言无关的实现途径: 一是基于组件对象模型 COM^[3] 的 ActiveX 控件技术; 二是动态链接库 DLL. 传统的 DLL 一般作为函数库^[4] 使用, 本文将 DLL 作为窗体对象库使用. 在 Delphi 中, 将窗体(模式窗体和非模式窗体) 添加至 DLL 中; Visual Basic 通过调用 DLL 中的导出过程直接使用窗体.

1 创建包含窗体的 DLL

实例程序: 在 Delphi 7.0 中, 分别创建包含日历组件(Tcalendar) 的模式窗体和非模式窗体, 并

将窗体添加至 DLL 项目中; 然后, 在 Visual Basic 应用中直接调用上述窗体.

1.1 模式窗体

包含模式窗体的 DLL 项目文件为:

library CalendarLib; //DLL 项目名称

uses //项目引用的单元:

SysUtils, Classes,

DllForm in 'DllForm.pas' {DllForm}; //窗

体文件

exports

ShowCalendar; //DLL 导出的过程

begin

end.

其中, ShowCalendar 的函数原型为:

function ShowCalendar (AHandle: THandle): TDateTime; StdCall;

Visual Basic 6.0 程序通过调用 ShowCalendar 函数来显示日历窗体(见图 1 所示), 当选择日历上的日期后, 该函数将日期返回给调用程序:

function ShowCalendar (AHandle: THandle): TDateTime;

var

DllForm: TDllForm; // TDllForm = class (TForm)

begin

Application.Handle := AHandle; // 将调用

收稿日期: 2003-11-18; 修订日期: 2003-12-30

基金项目: 国家自然科学基金资助项目(50099620); 河南省高校青年骨干教师基金资助项目

作者简介: 周振红(1963-), 男, 山东省蓬莱市人, 郑州大学副教授, 博士, 主要从事 GIS 教学和科研.

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

程序实例句柄赋给 DLL 应用对象

```
DLLForm := TDLLForm.Create ( Application -  
tion ); //动态创建窗体  
try  
    DLLForm.ShowModal ; //显示模式窗体  
    Result := DLLForm.cal DLLCalendar .Cal -  
endarDate ; // 返回日历控件日期  
finally  
    DLLForm.Free ; //释放窗体  
end ;  
end ;
```



图 1 Delphi 创建的窗体
Fig.1 Form created by Delphi

这里必须传递的一个参数是调用程序的实例句柄 (THandle) .DLL 中的应用对象和发起调用的应用对象是不同的,为了让 DLL 中的窗体能在调用程序中显示,必须把调用程序的句柄赋给 DLL 的 Application.Handle 属性.

1.2 非模式窗体

包含非模式窗体的 DLL 至于得提供两个过程:一个负责创建和显示窗体;另一个用于释放窗体.其过程原型分别为:

```
function ShowCalendar ( AHandle : THandle ) :  
Longint ;stdcall ;  
procedure CloseCalendar ( AFormRef : Longint ) ;  
stdcall ;
```

它们的实现体分别为:

```
function ShowCalendar ( AHandle : THandle ) :  
Longint ;  
var  
    DLLForm : TDllForm ;  
begin  
    Application.Handle := AHandle ;  
    DLLForm := TDllForm.Create ( Applica -  
tion );  
    Result := Longint ( DLLForm ); // 返回窗体  
    实例句柄  
    DLLForm.Show ; //显示非模式窗体
```

```
end ;  
  
procedure CloseCalendar ( AFormRef : Longint ) ;  
begin  
    if AFormRef > 0 then //检查窗体句柄  
        TDllForm ( AFormRef ).Release ; //释放  
    end ;
```

其中>ShowCalendar 函数和模式窗体中的同名函数的功能类似:把传入的应用程序句柄赋给 DLL 的应用句柄,然后创建窗体.所不同的是:该函数并没有释放窗体,而是返回长整型的窗体实例句柄,以便调用程序可以使用这个句柄来维护所创建的窗体.另外,Delphi 的函数调用与 Visual C++ 的相比有点特别:当调用执行至结果赋值语句 (Result) 时,并不立刻从函数中返回,除非结果赋值语句是函数体的最后一条语句;而 Visual C++ 的函数调用遇return 语句即返回.

CloseCalendar 过程先检查窗体句柄的有效性,然后调用Release 方法释放窗体.此处,发起调用的应用程序必须把ShowCalendar 函数返回的窗体句柄回传给该过程.非模式窗体的 DLL 项目和模式窗体的类似,只是多了一个导出过程CloseCalendar .

2 Visual Basic 调用 DLL 中的窗体过程

Visual Basic 直到 6.0 版本还不能创建标准的 Win 32 DLL ,不过它可以使用 Windows API 和其他语言工具创建的 DLL .

2.1 调用模式窗体

在使用上面创建的 DLL 前,先将 DLL 置于系统路径中.示例 Visual Basic 应用可只包含一个窗体,并在该窗体中放置一个命令按钮,以引用 DLL 中模式窗体;放置一个标签,以显示从日历组件中选择的日期.

在窗体通用段中声明 DLL 的过程原型:

```
Private Declare Function ShowCalendar Lib "Cal -  
endarLib .DLL" ( _
```

```
ByVal Handle As Long) As Date
```

对该过程的调用为:

```
Private Sub GetCalendar _Click()  
    Lbl = ShowCalendar ( App .PrevInstance )  
End Sub
```

在声明 DLL 过程原型时,需特别注意过程的实现机制(函数和子程序)和过程名称大小写的精确匹配,以及参数类型和参数传递方式(值传递和

引用传递)的对应.就参数传递而言,Delphi 缺省为传值,而 Visual Basic 缺省为传地址,要使后者也能以传值方式进行传递,须在参数前添加ByVal 关键字.

在调用 ShowCalendar 函数时,传递的应用程序句柄(长整型)是应用对象的PrevInstance 属性,而不是hInstance 属性.这一点和Delphi 不同,Delphi 传递的是应用对象的Handle 属性.

2.2 调用非模式窗体

示例 Visual Basic 应用可只包含一个窗体,并在该窗体中放置两个命令按钮,分别用来显示窗体和关闭窗口体.

(1) 在窗体通用声明段中声明 DLL 中的过程原型及窗体级变量.

```
Private Declare Function ShowCalendar Lib "Calendar.MLib.DLL" ( _ByVal Handle As Long) As Long
```

```
Private Declare Sub CloseCalendar Lib "Calendar.MLib.DLL" ( _ByVal Handle As Long)
```

```
Private FHandle As Long //用于存放 DLL 的窗体句柄
```

(2) 在窗体加载事件中初始化窗体句柄.

```
Private Sub Form_Load()
```

```
FHandle = 0
```

```
End Sub
```

(3) 显示非模式窗体.

```
Private Sub ShowCalendar_Click()
```

```
If FHandle = 0 Then //保证只显示一份窗体实例
```

```
FHandle = ShowCalendar (App.PrevInstance)
```

```
End If
```

```
End Sub
```

(4) 释放窗体.

```
Private Sub CloseCalendar_Click()
```

```
If FHandle <> 0 Then //先判断窗体
```

实例句柄的有效性

```
CloseCalendar (FHandle)
```

```
FHandle = 0
```

```
End If
```

```
End Sub
```

3 结束语

在 Windows 平台下,DLL 不仅可作为函数库使用,也可作为窗体对象库使用.将 Delphi 窗体扩展至 Visual Basic 应用中,须在显示窗体的 DLL 导出过程中设置长整型实例句柄参数,以传递调用程序的应用对象句柄,且 Visual Basic 应用传递的是应用对象的PrevInstance 属性,而不是hInstance 属性;若要显示非模式窗体,DLL 中显示窗体的导出函数须返回窗体实例句柄,以便调用程序可以使用这个句柄来维护所创建的窗体.

参考文献:

- [1] TEIXEIRA Steve,PACHECO Xavier.Delphi 6.0 开发人员指南[M].龙劲松,王瑜,谢尚书,译.北京:机械工业出版社,2003.
- [2] JERKE Noel.Visual Basic 5.0 开发人员指南[M].京京翻译组,译.北京:机械工业出版社,1997.
- [3] 周振红,周洞汝,杨国录.基于COM 的软件组件[J].计算机应用,2001,21(3):6~8.
- [4] 周振红,宋宇伟,郭恒亮,等.Visual Fortran 基于 Win32 DLL 的混合编程技术[J].郑州大学学报(工学版),2003,24(3):10~13.

Delphi Forms Being Applied to Visual Basic

ZHOU Zhen-hong¹, GUO Heng-liang¹, YANG Guo-tu²

(1. College of Environmental & Hydraulic Engineering, Zhengzhou University, Zhengzhou 450002, China; 2. College of Water Resources & Hydropower, Wuhan University, Wuhan 430072, China)

Abstract : Delphi forms are applied to Visual Basic in order to utilize Visual FormInheritance of Delphi and to supply a project group with flexibility in choosing development tools . The way is to add forms to a DLL project in Delphi , and to export routines from DLL , which are responsible for creating , showing and releasing forms , then to import routines in Visual Basic . The key is to set up a handle parameter to accept caller 's instance handle in exporting routines ; and to return a form instance handle used to close the form by caller if the form to show is modeless .

Key words : Delphi ; visual form inheritance ; visual basic ; DLL