

VB 与 MapInfo 的交互集成技术

方红雨 (合肥轴承总厂微机房 230033)

崔逊学 (中国科技大学自动化系 230027)

摘要: 本文介绍 VB 应用程序与 MapInfo 进行交互集成的技术细节, 并以例来演示具体的调用和回调实现过程。

关键词: VB MapInfo 对象链接 交互

地理信息系统(GIS)是以计算机图形图像处理、数据库技术和空间分析方法为基础, 处理一切与地理空间分布有关的资源、环境等多种信息的计算机空间数据管理与分析系统。GIS 作为一种电脑测图和分析工具, 将普通的数据库操作功能与地图所具有的直观分析功能相结合, 使其具有与其他应用系统不同的特点, 成为分析、预测和制定计划的重要手段。

现在 GIS 软件平台大多提供大量的功能指令和自己的编程环境, 同时以多种方式和其他应用程序进行交互, 因此可利用现有的程序设计语言(VB、VC、DELPHI 等)来控制平台的运行, 充分利用程序设计语言的强大功能, 根据需求来设计应用系统。本文介绍如何实现世界标准桌面地理信息系统 MapInfo Professional 与 VB6.0 应用程序之间的交互集成技术细节。

1 交互集成的技术原理

利用 VB 开发的 GIS 应用程序, 一般以 MapInfo 作为后台服务器来提供地理空间数据和管理图形化的地理数据, 并直观显示结果; VB 作为前台开发工具, 负责处理响应事件、控制 MapInfo 和存取数据等。由于 MapInfo 在后台进程作为本地服务器运行, 进行地图数字化, 编辑和查询图形与属性数据, 可大大提高数据访问速度。通过 VB 的 OLE 嵌入并生成 MapInfo 对象, 利用回调函数进行前后台的信息交换。

OLE 自动化技术在交互集成中起着关键作用, OLE 是软构件集成技术的基础。OLE 自动化技术是 OLE2.0 新增加的技术, 它使用户通过编程在一个应用程序中控制另一应用程序的对象, 从而实现应用程序级的可重用。由于

VB 和 MapInfo 都对 OLE 自动化技术提供良好的支持, 因此使用 OLE 的好处不仅在于 MapInfo 处理过的地图对象在 VB 中是可见的, 可用 VB 编程工具来访问它, 而且还能利用 MapInfo 的回调机制。

通常的位置信息、拓扑关系等空间数据和其他说明数据、属性数据等非空间数据应分别存放在不同的数据库中, 如果需完整处理某位置的所有信息, 则须对存放在不同数据库中的空间数据和非空间数据进行连接。数据交互的连接是通过一定的标识码进行的(如图 1 所示)。

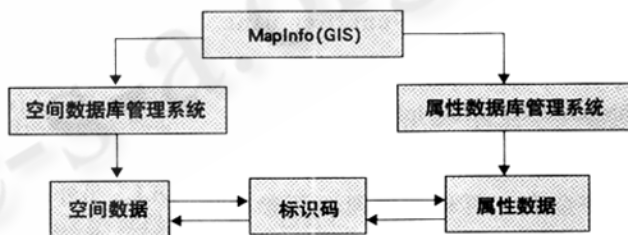


图 1 数据交互的连接结构

2 VB 应用程序调用 MapInfo

MapInfo 提供了对象链接和嵌入(OLE)方式来与其他软件链接, 如 MapInfo 的矢量化地图可容易嵌进应用程序, 并能随时激活 MapInfo 来编辑地图。OLE 把 MapInfo 作为一个对象嵌进容器(如 Word、VB 的应用程序), 应用程序通过对该对象方法的调用来启动、控制 MapInfo 的运行, 并通过应用程序的界面来观察地图的显示和操作的结果。两者在各自的线程内运行, 形成交互的主从结构, 这里应用程序是客户端, MapInfo 是服务器端。MapInfo

的地图显示到应用窗口,因此如果通过命令在MapInfo内装载了一个窗口,MapInfo将由于不能接受到重画消息而不能进行窗口刷新。

通过VB的OLE AUTOMATION机制,将MapInfo作为对象链接到应用程序,MapInfo作为本地服务器在后台运行,应用系统作为客户端在前台运行。因此应用系统既可通过发送命令来利用MapInfo管理和操纵地图化的数据,又可充分利用VB操作各种数据库,提供人工交互的界面操作。

现以一图形显示举例,程序的主窗口是MainForm,窗口上放置一名为MapPicture的PictureBox图像显示控件,则VB调用MapInfo的启动程序如下:

```
Public MapApp As Object '定义一名为MapApp的MapInfo对象
bUnableToCreateObj=False
On Error GoTo cant_createApplicationObject
Set MapApp=CreateObject("MapInfo.Application")
'生成MapApp对象,以后通过该对象操纵后台的MapInfo
If bUnableToCreateObj Then
    On Error GoTo cant-createRuntimeObject
    Set MapApp=CreateObject("MapInfo.Runtime")
    '如果MapInfo正在运行,则生成MapApp实时对象
End If
Load MainForm
MainForm.Show
Exit Sub
cant_createApplicationObject:
    bUnableToCreateObj=True
    Resume Next
cant_createRuntimeObject:
    MsgBox "对象生成错误"
    以打开某地图表文件的图层为例,调用MapInfo的连接程序段代码示例如下:
Set NowWindow=New MapWinClass
NowWindow.TFormName=MainForm.name
TablePath=App.Path+""',设置文件目录
MapApp.Do "Set Application Window" & MainForm.hWnd
MapApp.Do "Set Next Document Parent "& MapPicture.hWnd & " Style 1"
MapApp.Do "Open Table" & """" & TablePath & "MyTab.tab" & """" & "ReadOnly Interactive"
' 打开MapInfo中名为MyTab.tab的表文件
```

MapApp.Do "map from MyMap"
'打开MapInfo中MyTab.tab表文件的MyMap图层

MapInfo图形显示在MapPicture控件上,用户通过此PictureBox来观察地图,因此可利用VB强大的编程功能来操作对象MapApp,向后台的MapInfo发送命令,在当前窗口观察MapInfo执行命令后的地图显示。

链接的MapInfo不仅可完成单条命令,还能调用MapInfo的MBX模块,实现复杂的功能。该模块用MapInfo提供的MapBasic语言编写,可实现高级语言的绝大部分功能,并可向应用系统发送部分数据。若有一名为MyMbx.MBX文件供应用程序调用,则使用如下调用方式:

```
MapApp.Do "Run Application" & """" & AppPath & "/" & MyMbx.MBX" & """"
```

需补充说明的是,Do(string)是MapInfo实例对象的最常用方法之一,负责向MapInfo发送命令字符串(MapBasic语句),MapInfo将执行这些命令。

3 MapInfo的回调机制

MapInfo对象具有提供回调机制的功能。所谓回调就是MapInfo的某个事件发生引起MapInfo去调用客户端(VB应用程序)的某一个子程序。该子程序在客户端编写,但被MapInfo调用,被称为回调函数。MapInfo的按钮事件、菜单、鼠标事件都可以引起回调。通过回调函数就可以自动地向前台客户端发送消息,反过来客户端(应用程序)又可通过对象的接口操纵MapInfo。考虑到回调函数的程序代码数量和消息的一致性,可通过回调函数向应用程序的主窗口发送消息,主窗口利用回调函数获得的数据向用户提供执行结果。实质上此时应用程序应看成是服务器,MapInfo作为客户端,当MapInfo自身发生变化时,调用VB应用程序,通知自己的改变。

实施回调的步骤是首先在一个类中定义回调函数,再创建一个回调对象,当MapInfo的链接发生时为MapInfo创建按钮、菜单等,用对象的SetCallBack方法把回调函数设置为消息处理函数。因此当MapInfo图层上有事件发生时,回调对象就可自动调用相应的回调函数。SetCallBack(Idispatch)方法的功能是将OLE自动化对象登记为一个接受器,当MapInfo窗口变化时,由MapInfo通知应用程序,它们之间的关系如下:



(下转第76页)

例如使用回调机制来获取图层点击处位置和两点间距离,方法是首先在 Class Modules 中定义一回调函数 MapCallback(MapCallClass.cls),其包括两个子过程调用,即 Public Sub GetSelectPoint(ByVal CmdStr As String)与 Sub GetLength(ByVal CmdStr As String);然后在 VB 中需要使用这两种操作的地方进行回调功能处理,实施回调的关键程序段如下:

Public GCallBack As Object '定义一名为 GCallBack 的回调对象

MapApp.Do "Set Application Window " & MainForm.Hwnd
Set GCallBack=New MapCallback

MapApp.SetCallBack GCallBack '获取回调函数

MapApp.Do "Create Buttonpad " "ICallBack" "As Toolbutton
ID 2002 DrawMode 34 Cursor 0 Calling OLE " "GetSelectPoint
"" '获取点击处的位置

MapApp.Do " Create Buttonpad " " WCallBack "" As
Toolbutton ID 2003 DrawMode 36
Cursor 138 Calling OLE ""GetLength "" "" '获取两点间距离
和表中位置

从以上过程可看出,回调机制使得当 MapInfo 的窗口发生有关变化时,能够告之 VB 应用程序调用相关的应用功能模块函数进行实时响应和处理,甚至完成很多强大的功能如数学模型处理、预测决策等。■

参考文献

- 1 John W Fronckowiak 著,全刚等译,Visual Basic 6数据库编程大全,北京 电子工业出版社,1999
- 2 邬伦等,地理信息系统教程,北京大学出版社,1994
- 3 MapInfo Professional User's Guide,MapInfo Corporation,Troy,New York
- 4 MapInfo Professional Reference,MapInfo Corporation,Troy,New York