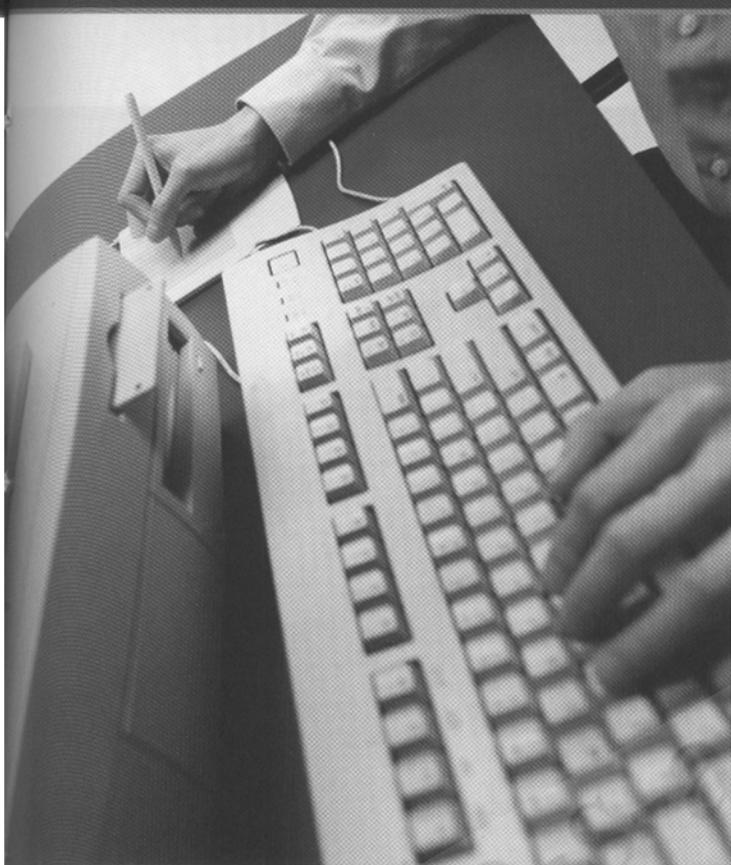


可视化编程语言 实现 GIS 软件的二次开发

李东平 赵军

(兰州 西北师范大学 GIS 开发应用研究中心 730070)



摘要: 本文比较了应用型 GIS 的实现方式, 并介绍一种基于 OLE Automation/DDE 技术实现应用型 GIS 的方法, 并以 MapInfo 和 Visual Basic 为例, 给出了系统实现的关键步骤及技术难点的解决方案。

关键词: GIS OLE Automation DDE 回调技术

地理信息系统是 60 年代发展起来的对空间信息进行分析与研究的一门新技术。进入 21 世纪以来, 随着数字地球概念的提出和社会信息总量的急剧膨胀, GIS 产业得以迅猛发展, 可观的发展前景吸引了众多的 IT 厂商, 纷纷推出自己的 GIS 软件产品。GIS 软件按功能和内容可以分为工具型 GIS 和应用型 GIS。前者是一种通用型软件, 具有 GIS 的一般特点, 但没有地理空间实体, 典型的如 MapInfo、Arc/Info 等, 后者是在 GIS 开发平台的基础上, 经过二次开发得到的适用于特定目的的系统。对于大多数用户来说, 并不需要功能复杂、难于掌握的工具型 GIS, 而是需要满足一定工作需要的应用型 GIS。所以, 市场上对应用型 GIS 的需求大大超过工具型 GIS。应用合适的开发工具对工具型 GIS 进行二次开发, 是得到既能满足需要又具有方便美观界面的应用型 GIS 的重要途径。

1 GIS 软件二次开发实现方式

1.1 宏语言开发

当今流行的 GIS 软件, 大多提供了用于系统二次开发的宏语言。如 ESRI 的 ArcView 提供了 Avenue 语言, MapInfo 公司的 MapInfo Professional 提供了 MapBasic 语言等等。系统开发者以 GIS 工具软件为开发平台, 利用宏语言开发针对不同应用对象的应用程序, 这类宏语言通常提供较强的地理分析功能, 但在用户界面开发和多媒体集成方面仍强差人意。作为编程语言, 与传统的通用编程语言相比有较大差距。

1.2 集成控件开发

一些 GIS 软件生产厂家也提供建立在 OCG 技术基础上的 GIS 功能控件, 如 ESRI 的 MapObjects、MapInfo 公司的 MapX 等, 在 Visual Basic 等编程环境下, 可直接将 GIS 功能嵌入到应用程序中, 从而实现地理信息系统的各种功

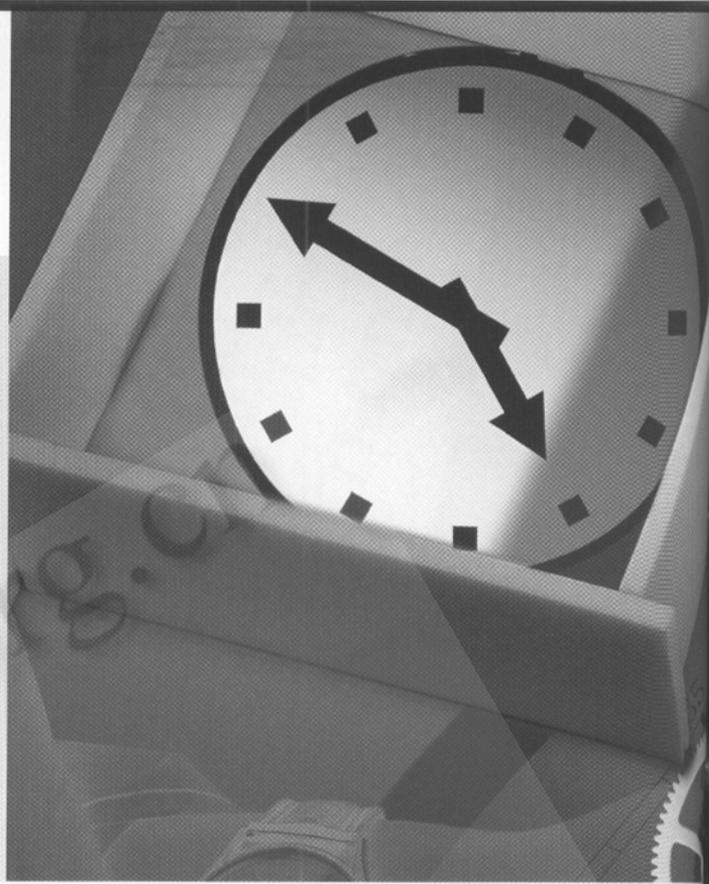
能。目前可供选择的开发环境很多, 除 Visual Basic 外, 还有 Visual C++, Visual FoxPro, Borland C++, Delphi, C++ Builder, Power Builder 等等。集成控件开发技术, 开发者不需要掌握 GIS 宏语言, 而是利用自己熟悉的编程语言进行系统开发, 但 GIS 控件往往只包括原系统平台的部分功能, 一些复杂的地理分析难以实现。

1.3 OLE/DDE 集成开发

OLE Automation (也称 OLE 自动化) 技术, 是整个 ActiveX 技术的核心。该技术能使对象在应用程序间方便地共享。目前大多数的 GIS 软件对外界提供 Automation 对象。DDE (Dynamic Data Exchange, 动态数据交换) 是一种通过应用程序共享的公共内存空间实现其相互通信的手段, 它允许支持 DDE 的应用程序之间通过连续自动地交换数据进行会话。OLE Automation 与 DDE 技术原理相似, 在应用型 GIS 开发中, OLE Automation 与 DDE 方式语法很接近。

采用 OLE Automation/DDE 技术, 利用软件开发工具开发前台可执行应用程序, 启动 GIS 工具软件在后台执行, GIS 工具软件作为一个后台服务器, 软件开发工具向后台的 GIS 工具软件发送命令, 并利用回调技术动态获取其返回信息。开发者可以使用自己熟悉的编程工具, 如 Visual Basic、Visual C++、Delphi 等实现应用程序中的地理信息处理功能。利用 OLE/DDE 集成开发, 几乎可以实现 GIS 工具软件的所有功能, 并可以利用可视化编程工具的强大功能实现精彩的操作界面, 采用这种方式, 开发周期短, 难度小, 是 GIS 二次开发的理想方式。

Realization of secondary development for GIS by visible programming language



2 OLE/DDE 集成开发的实现

这里以国内用户群较大的 GIS 工具软件 MapInfo Professional 和常用编程工具 Visual Basic 为例, 讨论 GIS 软件 OLE/DDE 集成开发的实现方法。

2.1 Visual Basic 后台启动 MapInfo Professional

声明一个 Visual Basic 的全局对象变量, 利用 Visual Basic 的 CreateObject() 函数启动 MapInfo,

```
Public olemap as object
```

```
Set olemap = CreateObject("MapInfo.application")
```

如果使用 DDE, 在 VB 中可以使用 Shell() 函数启动 MapInfo, 然后使用 LinkMode 建立与 MapInfo 的 DDE 连接。

2.2 前台工具向 MapInfo 发送命令

使用 OLE Automation 方法集成开发, 在启动 MapInfo 后, 可以利用“对象变量.do”的方法向后台的 MapInfo 发送字符串命令, 以实现各种 GIS 功能。

```
Olemap.do "Set Application Window "& Form1.
```

```
hwnd
```

```
Olemap.do "Set next Document Parent "& Form1.
```

```
hwnd & "Style 1
```

```
Olemap.do "Open Table "" C:\水系""
```

```
Olemap.do "map From 水系
```

如果使用 DDE 连接, 则利用 Visual Basic 的 LinkExecute 向 MapInfo 发送命令

2.3 定制 MapInfo 窗口

使用 DO 方法, 执行“Set Application Window”语句, 集成 MapInfo 的对话框和错误信息, 执行“Set Next Document”就可以在前台编程工具中集成 MapInfo 的窗口。同法还可以集成 MapInfo 的浏览窗口、信息窗口等其他窗口。在定制 MapInfo 窗口的过程中, “Set Application Window”和“Set Next Document”都必须指定一个控件的 Hwnd (句柄)。下一次创建一个 MapInfo 窗口时, 新创建的窗口就被重父化, 这样便于新建窗口将前台编程工具作为它的父类。

2.4 调用 MapInfo 菜单命令

开发过程中常常需要调用系统平台的标准菜

单命令。在 MapInfo 的 OLE/DDE 集成开发环境中, 使用 RunMenuCommand 语句实现调用, 如:

```
olemap.RunMenuCommand M-FILE-NEW
```

```
olemap.RunMenuCommand M-FILE-SAVE
```

上述语句可实现 MapInfo “文件/新建表”、“文件/保存表”菜单的调用。利用 RunMenuCommand 语句还可以实现集成 MapInfo 的标准工具栏, 如实现 MapInfo 标准工具栏的“漫游”工具:

```
olemap.RunMenuCommand M-TOOLS-RECENTER
```

MapInfo 为集成开发提供了标准的程序头文件, 其中包含了 MapInfo 菜单的标识符, Visual Basic 编程使用的头文件为 Mapbasic.bas, C 语言编程使用的头文件为 Mapbasic.h。

使用 RunMenuCommand 语句, 可以调用原系统平台的所有标准菜单命令和标准工具, 在前台编程工具中实现原系统平台几乎所有的功能, 这也是 OLE/DDE 集成开发相对于其他开发方式的主要优势之一。

2.5 释放后台 MapInfo

通过 Visual Basic 启动 MapInfo 是利用

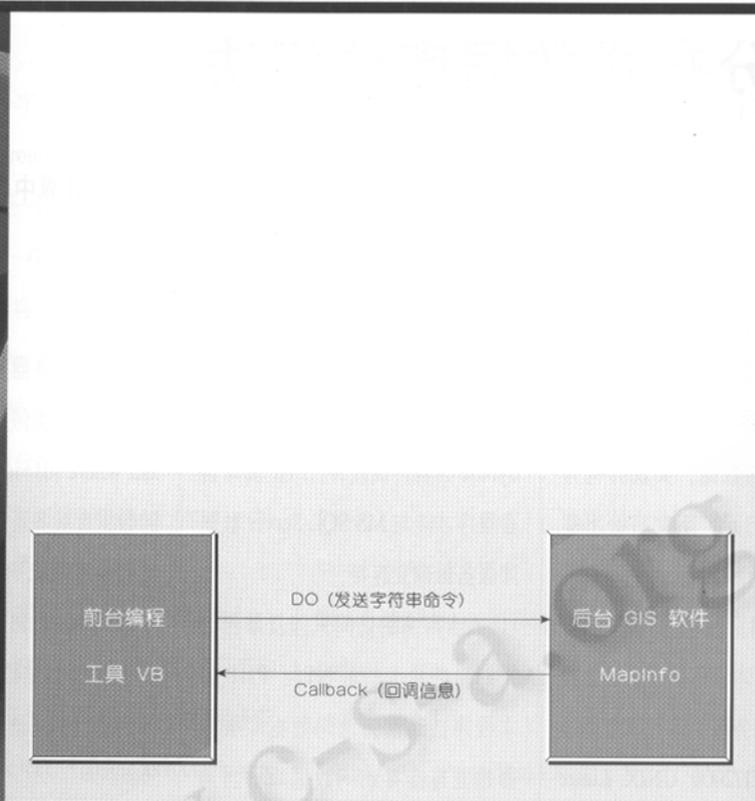


图 1 利用 Call back 回调方式

CreateObject() 函数调用一个对象变量，如果释放这个变量，后台的 MapInfo 也将自动终止。可以通过给对象变量赋一个 Nothing 值的方法来释放变量：

```
Set Olemap = Nothing
```

如果使用 DDE 方法与 MapInfo 连接，可以用 LinkExecute 方法发送一个“end Olemap”命令来终止 MapInfo。

3 技术难点

3.1 回调工作方式

采用 OLE/DDE 方法集成开发，是同时利用前台编程工具和后台 GIS 软件两个平台，平台之间实现相互传递信息是集成开发的关键环节。在 MapInfo 集成开发中，前台编程工具使用“对象变量.do”的方法向后台的 MapInfo 发送命令，MapInfo 向前台编程工具发送消息可以利用 Callback (回调) 过程来实现，如图 1。

如果使用回调技术，前台编程工具必须可以充当 DDE 或 OLE 服务器。目前主要流行的编程工具大多可以实现此功能，但早期版本的编程工

具很多只能做 DDE 服务器，而不能担任 OLE 服务器，例如 Visual Basic 4.0 或更高版本，才能充当 OLE 服务器。所以，如果使用较早版本的编程工具，最好采用 DDE 方式。

3.2 实现回调

实现回调的方法是：先在 Visual Basic 的类模块 class modules 中定义一个回调函数 mapcallback.cls，其中包含用于处理返回 MapInfo 信息的过程，在 Visual Basic 中需要使用该操作时进行回调处理。下面简要列出程序段：

```
Public olecallback as object '定义一回调对象
```

```
Set olecallback = New mapcallback '程序中  
使用回调
```

```
Olemap.SetCallback olecallback
```

```
Olemap.do"定义按钮语句" Calling OLE  
"mapcallback.cls 中定义的过程名"
```

这样就实现了一个标准的回调过程。

如果使用 DDE 方法，则可以执行 Calling DDE "MapInfo"，"窗体" 语句实现。可以看出在 MapInfo 窗口发生变化时，可以利用回调机制告之前台编程工具调用相关的应用功能模块进行

实时响应和处理，进而进行复杂功能的实现，如专家模型建立、数据即时分析等。

与其他二次开发方法比较，OLE Automation/DDE 方法有较大的优势。随着 OLE Automation/DDE 技术不断发展和 GIS 软件的完善，应用 OLE Automation/DDE 技术实现应用型 GIS 软件将更加简单、便捷。 ■

参考文献

- 1 龚健雅，当代 GIS 的若干理论与技术，武汉测绘科技大学出版社，1999。
- 2 方红雨，VB 与 MapInfo 交互集成技术，计算机系统应用，2001(5)。
- 3 于海龙，MapInfo 环境下二次开发相关问题的研究，东北测绘，2000(3)。
- 4 修文群，地理信息系统 GIS 数字化城市建设指南，北京希望电子出版社，2001。