

文章编号: 1005-0523(2008)03-0064-05

基于 OLE 自动化技术的 SPSS 二次开发原理及应用

胡 辉

(华东交通大学 经济管理学院应用统计研究所 江西 南昌 330013)

摘要: 随着计算机技术的迅速发展, 建立在各种统计方法模型基础上的统计分析软件在实际中的应用成为可能. 但如何灵活使用统计软件, 深度挖掘软件功能去解决实际中的复杂问题, 往往需要对软件进行二次开发. 首先从软件开发原理的角度, 以常见的统计分析软件 SPSS 为例, 介绍了 SPSS 软件体系结构, 并深入分析了其支持二次开发的原理: 基于组件的 OLE 自动化技术. 最后以 Visual C++ 为例说明了如何在其他应用程序中使用 SPSS 提供的功能进行二次开发.

关键词: SPSS; 二次开发; 组件; 对象链接与嵌入; 自动化; Visual C++

中图分类号: TP312

文献标识码: A

随着计算机技术的迅速发展, 传统的很多统计方法在实际上的应用成为可能, 各种建立在统计方法模型上的统计分析软件不断涌现. 如国外的 SAS, SPSS, STATISTICA 等, 国内如 DPS 数据处理系统, NOSA(非典型数据统计分析系统), 马克威 Markway 统计分析与数据挖掘软件等. 然而, 由于实际工作中问题的复杂性, 往往难以做到原始数据恰好符合软件中的统计模型和方法的适合条件(这需要对统计方法模型有正确的理解), 或是在其他应用系统(如 Excel 甚至是一些决策支持系统)中希望利用统计软件得出的数据进行再加工(这需要对统计软件开发原理本身有深入的了解). 因此, 灵活使用统计软件、深度挖掘软件功能就变得必要, 而建立在这些统计分析软件基础上的二次开发便是解决这些问题的主要手段.

事实上 SAS, SPSS, STATISTICA 这些具有国际影响力的统计分析软件在支持二次开发方面是非常强大的. 如 SAS 本身就主要面向具有专业编程能力的统计人员; SPSS 一方面依靠菜单、对话框的方式赢得了非统计专业人士的厚爱, 另一方面它内置的语法 Syntax、脚本 Script、SaxBasic 语言对于欲灵活使用它的人带来了很大的便利; STATISTICA 中的 STATISTICA Visual Basic 语言也同样是如此.

鉴于 SPSS 适用面的广泛性(既针对统计专业

人员, 又面向非统计专业人士), 在此以 SPSS 为例探讨二次开发的技术原理及其应用. 首先从软件构建角度分析 SPSS 软件体系结构, 然后着重在技术开发原理上深入探讨软件支持二次开发的原理, 最后以 Visual C++ 为平台说明如何应用软件中的对象模型进行二次开发.

1 SPSS 软件的体系结构^[1]

基于对象、基于组件的软件体系目前是主流的软件架构, SPSS 软件体系也莫能外. 它本身就是建立在面向对象、组件的基础之上的, 其中 SPSS 类库和对象是支撑 SPSS 整个软件体系的基础.

1.1 SPSS 类库

类库是文件或文件中的组件, 主要有两种类型(.tlb、.olb)的类库. 扩展名为.tlb的类库可以作为单独的文件进行安装, 扩展名为.olb的类库可以嵌入到对象库文件内部. SPSS 提供了 SPSS 类库(spss-win.tlb)、SPSS 转轴表类库(spssvt.tlb)、SPSS RTF 类库(spssrtf.tlb)和 SPSS 图形编辑器 OLD 控制库(spssgctl.tlb).

安装完 SPSS 软件以后, 第一次运行它时 SPSS 类库会自动注册到 Windows 注册数据库中, 以后若有其他应用程序使用 SPSS 类库中的对象便可以直

收稿日期: 2008-03-19

作者简介: 胡 辉(1973-), 男, 江西萍乡人, 华东交通大学经济管理学院统计系教师, 研究方向为统计分析与数据挖掘.

接从中访问。

1.2 SPSS 对象

对象是具有各种数据分析功能的独立实体,可以通过程序进行控制。SPSS 提供了数十个对象,包括简单的文本对象、图例对象和比较复杂的转轴表对象、交互图对象等,其中实现 SPSS 基本功能的对象包括 Application 对象、Options 对象、Documents Collection 对象、Data Document 对象、Syntax Document 对象、Draft Document 对象、Chart 对象、OutputItems Collection 对象、Output Item 对象和 Map 对象等。

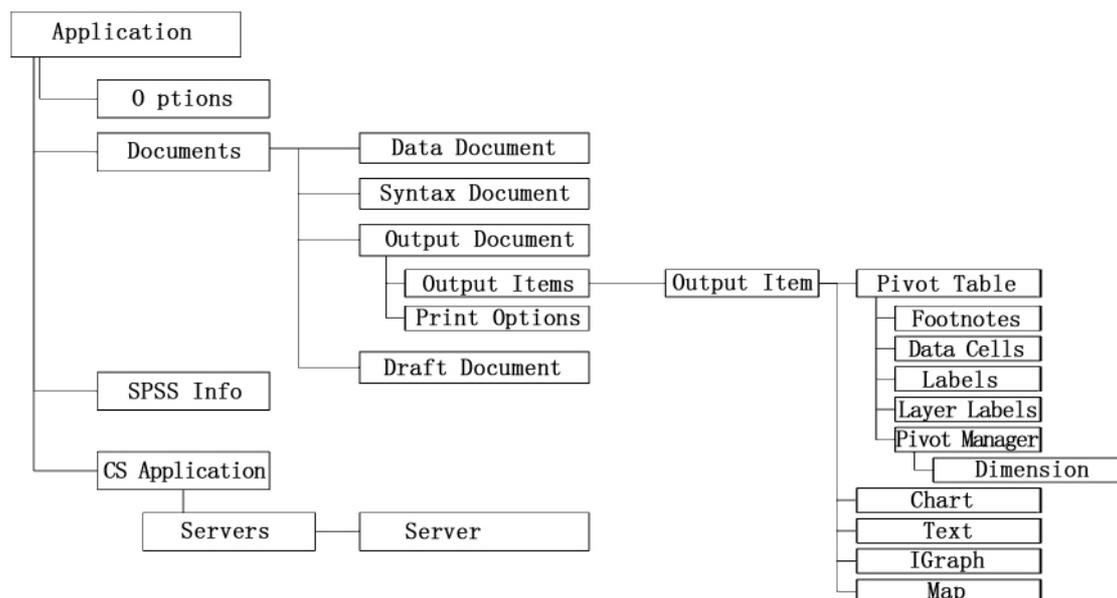


图1 SPSS 对象的层次结构图

2 二次开发的技术原理: 基于组件的 OLE 自动化技术^[2]

一个软件若需要在其基础上进行二次开发,与其他应用程序进行通信集成(如 Office 中的宏、脚本语言)或与其他语言有外部接口,则一般需具备 OLE 自动化功能。目前很多应用程序都具有 OLE 自动化功能,如 Microsoft Office 中的 Word、Excel、AutoCAD、MapBasic、SPSS 等。OLE (Object Linking and Embedding) 意为对象链接与嵌入,是微软对象技术的名字,其本质是一种通信技术,允许一个应用程序访问另一个应用程序的内部结构。目前的 OLE 基础是组件技术,而非早期的 DDE (Dynamic Data Exchange) 动态数据交换。

2.1 组件技术原理

面向软件复用的组件技术实质是在二进制代码

图 1 是 SPSS 中对象的树形结构,位于最顶端的是 Application 对象,它代表 SPSS 自身,所有其他对象都在 Application 对象之下。第二层包括 4 个对象:

- 1) Option 对象: 利用该对象可以为输出浏览器、图表和数据等设置选项。
- 2) Documents 对象: 提供数据编辑、语法文档窗口、输出浏览器和草稿文档窗口等 4 个窗口的属性和操作方法。
- 3) SPSS Info 对象: 提供 SPSS 的一些必要信息。
- 4) CS Application 对象: 提供与网络有关的对象操作。

上实现软件的复用,它超越了结构化开发思想的函数复用和模块复用,也与面向对象思想的类的复用有本质不同。组件通过对一组类的组合进行封装,隐藏了具体的实现,只用接口提供服务,从而在不同层次上,组件均可以将底层的多个逻辑组合成高层次上的粒度更大的新组件,甚至直接封装到一个系统,使模块的重用从代码级、对象级、架构级到系统级都可能实现。要理解应用组件技术,必须深入体会组件的思想:

- (1) 组件是在二进制一级的,而传统的面向对象是源代码一级的。
- (2) 接口是组件最为重要的特性之一,接口是一组函数,通过它与其他程序通信。
- (3) 通过 GUID (Globally Unique Identifier) 全球唯一标识符来唯一标识组件类和接口。
- (4) 组件对象能够自己控制生存期,能支持多个接口,管理运行时发现功能。

(5) 组件产生是通过组件服务器产生的,分进程内和进程外两种。进程内是 DLL 形式,进程外则是 EXE 形式。

2.2 OLE 自动化技术

OLE 自动化是通过组件对象模型实现的,但组件接口使用的是早绑定的机制,也就是说这对编译语言较适合,但由于解释型的程序语言在运行时需要迟绑定,这个机制就会失效,因此需引入自动化和新的接口 IDispatch 及其中两个重要的新函数 Invoke、GetIDsOfNames,这是自动化技术的关键所在。

1) IDispatch 接口

自动化需要迟绑定的机制,这个机制是通过 IDispatch 接口实现的。譬如一个组件对象实现了 IDispatch 接口,那么像 VB 这样的程序就可以在 Add() 和 Subtract() 这两个函数进行迟绑定,如下图所示:

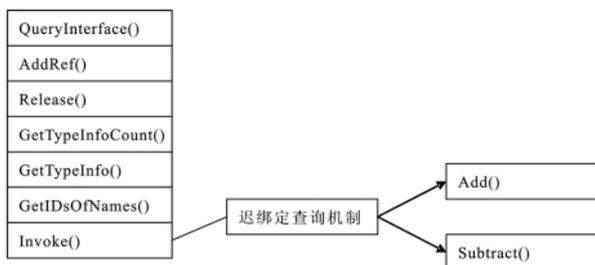


图2 IDispatch 迟绑定机制

从上图可以看出 IDispatch 也是一个 COM 接口,最上面放的是 3 个标准的 IUnknown 接口函数,后面则是 IDispatch 本身的函数。但是,Add 和 Subtract 两个函数不是直接通过应用程序的虚函数表实现的,而是通过 Invoke、GetIDsOfNames 函数来管理参数传递机制和函数指针的问题。

2) Invoke 函数

Invoke 函数是实现自动化技术的关键,它比较复杂,有 8 个参数,函数形式为: Invoke (DISPID dispID, REFIID riid, LCID lcid, unsigned short wFlags, DISPPARAMS * pDispParams, VARIANT * pVarResult, EXCEPINFO * pExcepInfo, unsigned int * puArgErr)。

第一个参数是 dispID,用来标识调用的自动化方法或属性。一般不必关心这个值到底是多少,只要得到这个变量,就能和系统进行通信。

第二个参数是 riid,指一个接口的引用。这个参数是保留的,永远都是 IID - NULL。

第三个参数是 lcid,表示解释参数所使用的当

前上下文环境,一般支持多语言环境使用它。

第四个参数是 wFlags,标识调用的是函数还是属性。

第五个参数是 pDispParams,用来完成函数调用是迟绑定还是早绑定。

第六个参数是 pVarResult,存放函数结果也就是返回值。

最后两个参数用于处理错误情况及错误代码。

3) GetIDsOfNames 函数

Invoke 只有通过 dispID 才能知道需要访问哪个属性或方法。在一些解释性语言如 VB 中是使用名字来得到 dispID 的, IDispatch 接口正是用一个这样的函数 GetIDsOfNames 把相关属性或方法和一组参数的名字映射为一组 DISPID,从而通过名字转化为 dispID。

3 SPSS 软件的二次开发^[3]

SPSS 软件的二次开发一方面是利用内置的 SaxBasic 和 Syntax 命令进行编程(在此不作介绍),另一方面则是如何与其他开发环境集成在一起,利用它提供的功能为别的应用系统服务。在此,主要以 Visual C++ 6.0 为例说明如何在 VC 环境中使用 SPSS 提供的描述统计功能:计算均值和标准差,基本思路是首先导入可能要用到的库文件,得到相应的 SPSS 对象;然后调用 SPSS 的函数功能,得到分析结果;最后从分析结果中取出所需要的数值,返回给主程序。

3.1 导入 SPSS 类库: spsswin. tlb、spssppvt. tlb

本程序只是计算均值和标准差,所以除基本类库 spsswin. tlb 外还需要转轴表类库 spssppvt. tlb,此类库中包含了与转轴表有关的对象。SPSS 中的描述统计分析结果就是一个转轴表对象,利用这个对象下的 Data Cells 就可以获取表中元素的值。

在 VC 中导入类库操作很简单:在 VC 开发环境中建立一个新的工程,打开菜单 View -> Class Wizard,选择 spsswin. tlb、spssppvt. tlb 则会在现有的工程中添加 4 个新的文件: spsswin. h、spsswin. cpp、spssppvt. h、spssppvt. cpp,同时看到新生成了不少 SPSS 的对象,如图 3 所示。

可以看到这些 SPSS 对象的基类都是具有 OLE 功能的类 COleDispatchDriver,而实际上 COleDispatchDriver 本身就具有 IDispatch 接口的功能。

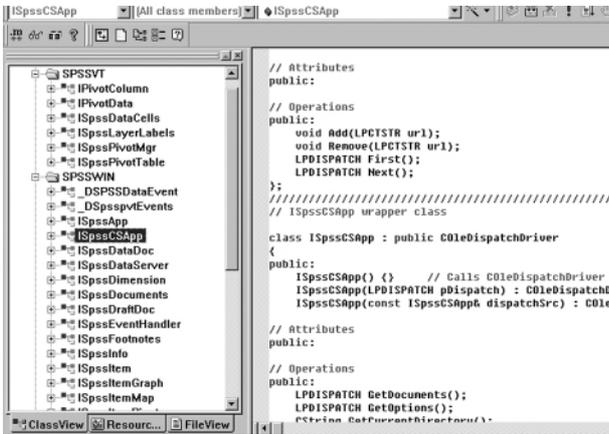


图 3 导入类库后新生成的 SPSS 对象

3.2 调用 SPSS 对象的功能

为说明如何调用 SPSS 对象的功能,以 SPSS 中的描述统计为例,建立一个简单的对话框,如图 4 所示。

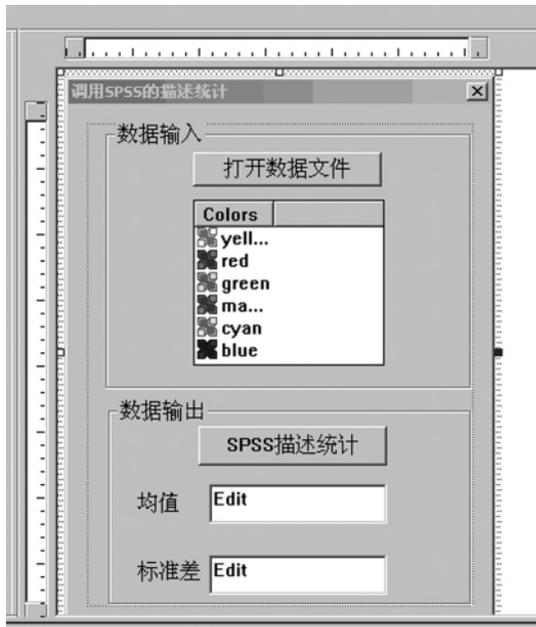


图 4 调用 SPSS 描述统计功能的对话框设计

对话框分两部分:一部分是数据输入,通过“打开数据文件”的按钮把数据文件中的数据输入到 List 控件中;另一部分是数据输出,通过“SPSS 描述统计”按钮实现结果输出到均值、标准差的 Edit 控件中。在“SPSS 描述统计”按钮中实现结果输出的主要代码如下:

```

theOutDoc = theDocs. GetOutputDoc ( 0 ); //获取
SPSS 输出结果
ISpssItems theItems;
theItems = theOutDoc. GetItems ( ) ; //获取输出
结果中的条目
    
```

```

ISpssItem theItem;
ISpssPivotTable thePivot;
int out = theItems. GetCount( ) ;
theItem = theItems. GetItem( out - 1 ); //获取输
出结果中的描述统计分析表
thePivot = theItem. ActivateTable( ) ;
ISpssDataCells theData; //获取描述分析表中的
数据单元
theData = thePivot. DataCellArray( ) ;
double f1 ,f2;
- variant - t t;
t = theData. GetValueAt( 0 ,1 ); //获取数据单元
中的“均值”
f1 = ( double) t;
m - Average = f1; //获取的“均值”赋给对话框
中的均值 Edit 控件相联的变量
t = theData. GetValueAt( 0 ,2 ); //获取数据单元
中的“标准差”
f2 = ( double) t;
m - Deviation = f2; //获取的“标准差”赋给对话框
中的标准差 Edit 控件相联的变量
UpdateData( false ); //获取后的数据返回到原
对话框中
    
```

3.3 输出实现结果

依照上述思路,经过编译运行,程序执行结果如图 5。



图 5 调用 SPSS 描述统计功能的结果输出

其中,数据输入是“打开数据文件”后列表框中“客流”的数据,总共有280个;数据输出中,均值为7910.632,标准差为12652.048,完全与SPSS实际输出结果一致,从而实现了利用SPSS功能完成有关统计分析的计算。

4 结束语

作为一个优秀的统计分析软件,SPSS在各个领域得到了广泛的应用。但如何更好地利用它去解决实际问题尤其是复杂的问题可能需要在其基础上进行二次开发。这意味着对应用者提出了更高的要求,因为一方面需要了解软件技术的开发原理,另一方面也需要对统计方法模型和实际问题有深刻的认识。从实现的方法角度看,文中介绍的开发思路和示

例主要是起借鉴启示的作用,如若真正要在实际中开发一些具体的应用,可能需要做更多的工作。譬如,不同的应用开发环境(如Excel、Visual Basic等)如何操作使用SPSS对象;另外如何获得更为复杂的统计模型方法(如多元回归分析、判别分析、因子分析等,而不像示例中的简单描述统计)的计算结果。

参考文献:

- [1] 苏金明. 统计软件SPSS12.0 for Windows应用及开发指南[M]. 北京: 电子工业出版社, 2004.
- [2] George Shepherd, Scot Wingo. 深入解析MFC[M]. 赵剑云, 译. 北京: 中国电力出版社, 2003.
- [3] 金雪云. Visual C++教程[M]. 北京: 清华大学出版社, 2005.

Principle of SPSS Secondary Development and Its Application Based on OLE Automation

HU Hui

(School of Economic Management, East China Jiaotong University, Nanchang 330013, China)

Abstract: With rapid development of computer technology, it's possible for statistics software which is built on various statistics models and methods to be put into practice. However, using statistics software smartly and maximizing its power to solve complex practical problems need a secondary development. Firstly in terms of software development principle, taking SPSS which is commonly used, SPSS software system is introduced and the principle of its secondary development which lies in OLE automation based on COM is analyzed thoroughly. Finally, an example developed in Visual C++ environment is demonstrated to realize SPSS functions in other applications.

Key words: SPSS; secondary development; COM; OLE; automation; Visual C++

(责任编辑:周尚超)

