

文章编号:1671-6833(2007)04-0066-04

Delphi 通过双接口控制 AutoCAD 绘图

赵峰¹, 毕苏萍², 周振红³

(1. 郑州大学 科研处, 河南 郑州 450001; 2. 郑州大学 土木工程学院, 河南 郑州 450001; 3. 郑州大学 环境与水利学院, 河南 郑州 450001)

摘要: 利用 AutoCAD 提供的编程模型, 可使其采取批处理方式高效绘图, 还可使应用程序拥有强大的图形功能. 编程控制 AutoCAD 绘图, 通常是 VB/VBA 利用其对象模型的自动化接口绘图, 其程序的执行性能相对较低. 现提出 Delphi 通过 AutoCAD 提供的对象模型的双接口进行绘图的方法, 并给出典型实例的具体实现. 实例结果表明: Delphi 绘图方法既提供了编程灵活性, 又使应用程序拥有较高的执行性能, 从而为编程控制 AutoCAD 绘图提供了一条新途径.

关键词: Delphi; AutoCAD; 编程模型; COM; 双接口

中图分类号: TP 311 **文献标识码:** A

0 引言

通常, AutoCAD 采取命令式的交互方式绘图. 若要批处理绘图或将 AutoCAD 作为服务器内嵌在客户程序中运行, 则需要编程实施, 而且多采用 Visual Basic/VBA 利用 AutoCAD 提供的自动化^[1]接口绘图, 其应用已十分广泛, 但 Delphi^[2-3]驱动 AutoCAD 绘图却很少被涉足.

事实上, AutoCAD 的编程模型提供了双接口 (VTbl 和 IDispatch), Delphi 作为客户程序, 则可充分发挥 AutoCAD 对象模型的双接口优势: 既提供了编程灵活性, 又提高了程序执行性能, 也为 Delphi 开发的应用程序提供了强大的图形功能.

笔者将对 Delphi 通过双接口控制 AutoCAD 绘图进行系统探讨, 并以典型实例加以说明.

1 AutoCAD 的对象模型

AutoCAD 提供的对象层次模型^[4] (如图 1 所示) 类似于 DAO 对象模型, 也是由根对象逐级向下引用. 其中, Document 对象对应于 AutoCAD 中的一幅图形. 控制 AutoCAD 绘图, 大多是在 Document 的 ModelSpace 中进行的.

AutoCAD 的对象模型基于 COM^[5-6], 当中的每一个集合/对象都实现了一个双接口, 且接口名由对象名加前缀 "I" 构成. 如 AcadDocument 对象

实现的默认接口为 IAcadDocument.

既然对象模型提供了双接口, Delphi 就可以通过 VTbl (V 表) 和 IDispatch (自动化接口) 两种方式使用 AutoCAD 提供的对象模型.

2 利用双接口绘图

2.1 VTbl 方式

要采取 VTbl 方式使用 AutoCAD 中的对象, 首先须在 Delphi 工程中引入 AutoCAD 的类型库, 并将产生的类型库单元 (AutoCAD_TLB. pas) 插入其中.

VTbl 方式实质是对基于 COM 的默认接口进行操作. 要连接到 AutoCAD 服务器, 可直接调用 AutoCAD 应用组件类的构造函数, 并返回其默认接口 IAcadApplication:

```
IApp := CoAcadApplication. Create; 或  
IApp := CoAcadApplication. CreateRemote ('RemoteMachine');
```

该操作将创建一个 AutoCAD 应用对象实例, 并产生一个 Document 对象. 所以, 接下来可引用其文档接口 IAcadDocument, 以进行后续的绘图操作:

```
IDoc := IApp. ActiveDocument;
```

编程控制 AutoCAD 绘图, 关键要处理好各种 Delphi 数据类型的数组. 在这里选取一个具有代表性的绘图操作: 画一个圆, 并在其中填充斜线.

收稿日期: 2007-10-07; 修订日期: 2007-11-10

基金项目: 河南省科技攻关资助项目 (0496050905)

作者简介: 赵峰 (1974-), 男, 郑州大学讲师, 硕士, 主要研究方向为计算机应用及图像处理.

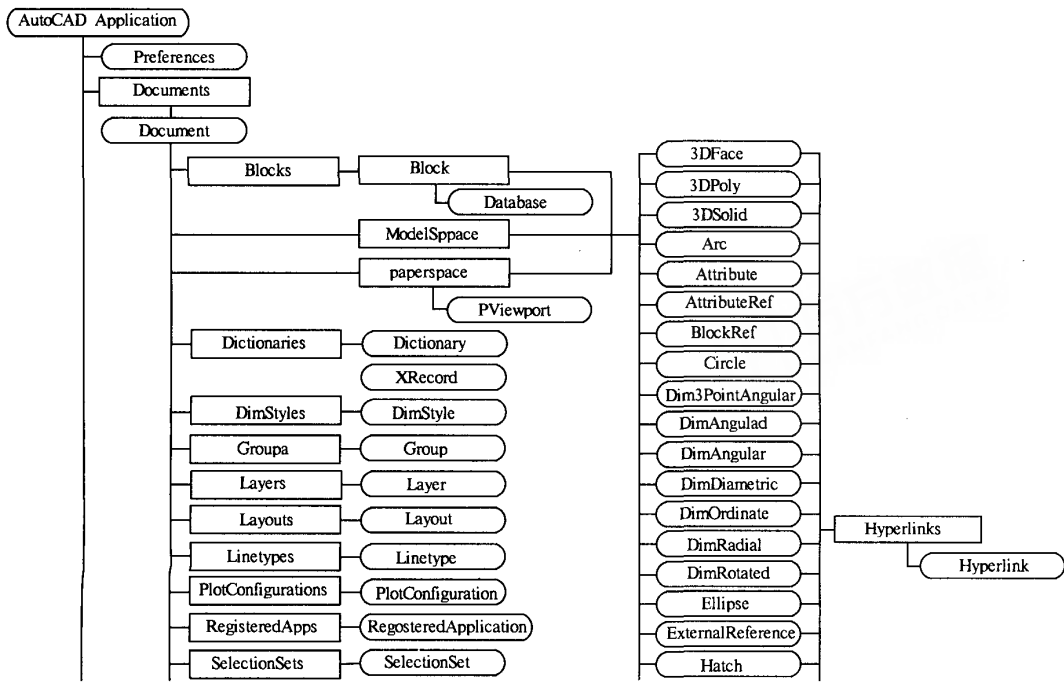


图 1 AutoCAD 的对象模型

Fig.1 The object model of AutoCAD

该操作既涉及实型数组,又包含实体对象数组. 下列为一完整的事件过程代码:

```
procedure TForm1. btnCreateHatchClick (Sender:
TObject);
```

```
var
```

```
    Hatch: IAcadHatch;
```

```
    OuterLoop: Array of IAcadEntity;
```

```
    Center: Array of Double;
```

```
    Radius: Double;
```

```
    PatternName: String;
```

```
    PatternType: Integer;
```

```
    bAssociativity: Boolean;
```

```
begin
```

```
    //Define and create the hatch
```

```
    PatternType := 0; PatternName := 'ANSI31';
```

```
bAssociativity := True;
```

```
Hatch := IDoc. ModelSpace. AddHatch ( PatternT-
ype, PatternName, bAssociativity );
```

```
    //Create a circle as the outer loop for the hatch
```

```
    SetLength( Center, 3 );
```

```
    Center[ 0 ] := 30; Center[ 1 ] := 30; Center[ 2 ] := 0; Radius := 10;
```

```
    SetLength( OuterLoop, 1 );
```

```
    OuterLoop[ 0 ] := IDoc. ModelSpace. AddCircle
```

```
    ( Center, Radius );
```

```
    //Append the outer loop to the hatch
```

```
    Hatch. AppendOuterLoop ( outerLoop );
```

```
    //Evaluate and display the hatch
```

```
    Hatch. Evaluate;
```

```
    IDoc. Regen( acTrue );
```

```
    IApp. Visible := True;
```

```
end;
```

因为组件类对象提供的服务,实质是由其实现的接口方法和属性提供的. 所以,在这里直接对 AutoCAD 的对象接口进行操作.

当中所涉及的数据,采取 Object Pascal 内建的基本类型或类型库单元所声明的类型或常量;所涉及的数组,统一采取 Object Pascal 的动态数组形式.

2.2 IDispatch 方式

IDispatch 接口为双接口当中的自动化接口. 在 Delphi 7 中,通过调用 ComObj 单元的 CreateOleObject 函数,来返回 IDispatch 接口.

使用 IDispatch 接口有两种方式:一是利用 as 运算符将 IDispatch 接口转换为组件类的 dispinterface 接口,然后再调用该接口的方法,其操作类似于前述的 VTbl 方式. 当然,这种方式需

要类型库的支持。

另一种方式是将 IDispatch 接口赋给一个 Variant, 以利用 Variant 内建的接口支持功能。在调用接口方法时, Variant 先取出接口 ID, 再引发 (invoke) 特定的方法。这种方式只利用了 AutoCAD 的系统注册表信息, 不需要类型库的支持; 但由于采取了运行时的动态绑定, 所以运行速度相对慢一些。

为便于比较, 这里给出上述示例的动态绑定代码:

```
procedure
TForm1. btnCreateHatch_DispatchClick ( Sender:
TObject );
var
  App, Doc, Hatch, Center, OuterLoop: Variant;
  Radius: Double;
  PatternName: String;
  PatternType: Integer;
  bAssociativity: Boolean;
begin
  //Assign the interface to a Variant
  App := CreateOleObject ( ' AutoCAD. Applica-
tion' );
  Doc := App. ActiveDocument;
  //Define and create the hatch
  PatternName := ' ANSI31 ' ; PatternType := 0;
  bAssociativity := True;
  Hatch := Doc. ModelSpace. AddHatch ( PatternType,
  PatternName, bAssociativity );
  //Create a circle as the outer loop for the hatch
  Center := VarArrayCreate ( [ 0, 2 ], varDoub-
le );
  Center [ 0 ] := 30; Center [ 1 ] := 30; Center [ 2 ]
:= 0;
  Radius := 10;
  OuterLoop := VarArrayCreate ( [ 0, 0 ],
varDispatch );
  OuterLoop [ 0 ] := Doc. ModelSpace. AddCircle
( Center, Radius );
  //Append the outer loop to the hatch
  Hatch. AppendOuterLoop ( outerLoop );
  //Evaluate and display the hatch
  Hatch. Evaluate;
  Doc. Regen ( True );
  App. Visible := True;
```

万方数据

end;

从中可以发现: 不但接口用 Variant 表示, 而且数组也采取了 Variant 数组形式, 这样便能满足与自动化兼容的数据类型要求。这里尤其要注意: AutoCAD 自动化对象的 VarType 须是 varDispatch。

3 与 AutoCAD 环境交互

通过 AutoCAD 对象模型提供的双接口, 不但可以控制 AutoCAD 绘图, 而且可以与 AutoCAD 的环境进行交互, 如设置和提取系统变量、访问 AutoCAD 的命令行、控制应用和绘图窗口、输入/输出其它文件格式等。和本文主题密切相关的是输出其它文件格式。

Delphi 控制 AutoCAD 绘图的完整解决方案应该是: 将 Delphi 作为应用系统的集成开发工具, AutoCAD 服务器作为绘图工具, 将 AutoCAD 幕后绘制的图形输出并载入 Delphi 窗体的图像组件中。除了执行速度稍慢一些, 用户根本感觉不到是 AutoCAD 绘制的图形。

下列示例代码将上述图形以 BMP 文件格式输出:

```
procedure TForm1. btnExportBmpClick ( Sender:
TObject );
var
  ISet: IAcadSelectionSet;
  MyEntity: Array of IAcadEntity;
  CurDir, ExportFile: String;
  I: Integer;
Begin
  //Fetch entities in the ModelSpace
  SetLength ( MyEntity, IDoc. ModelSpace. Count );
  For I := 0 To IDoc. ModelSpace. Count - 1 do
    MyEntity [ I ] := IDoc. ModelSpace. Item ( I );
  //Add entities to a SelectionSet
  ISet := IDoc. SelectionSets. Add ( ' NEWS-
SET' );
  ISet. AddItems ( MyEntity );
  // Export the drawing as a BMP file
  GetDir ( 0, CurDir ); { 0 = Current drive }
  ExportFile := CurDir + ' \Bmp\CadBmp' ;
  IDoc. Export ( ExportFile, ' BMP', ISet );
end;
```

输出 BMP 文件, 需提供一个选择集, 以规定从图形中输出的实体对象。由于这里是 VTbl 绑

定,所以采取了实体对象接口的动态数组形式,而不是变体数组形式。

4 结语

实例结果表明:Delphi 可通过 AutoCAD 编程模型提供的双接口来访问其对象服务。VTbl 方式需要类型库支持,操作所涉及的数组须采取动态数组形式;IDispatch 方式又分为 dispinterface 接口和 Variant 两种形式,前者需要类型库的支持,其操作类似于 VTbl 方式,后者则利用了 Variant 自身的接口支持功能,其操作类似于 VB 的动态绑定,但不需要类型库的支持,所涉及的数组须采取变体数组形式。因此,Delphi 通过双接口控制 AutoCAD 绘图的关键是正确使用 Delphi 各种数据类型的数组。

参考文献:

[1] 周振红,杨国录,周洞汝,等. AutoCAD 的 ActiveX

自动化技术[J]. 中国农村水利水电,1999,(6): 35-36.

[2] 周振红,李端有,谈戈. Fortran 与 Delphi 混合编程[J]. 计算机应用,2001,21(4):91-93.

[3] 周振红,郭恒亮,杨国录. 在 Visual Basic 中嵌入 Delphi 窗体的应用技术[J]. 郑州大学学报:理学版,2004,25(1):70-72.

[4] 郭朝勇,黄建波,王保平,等. AUTOCAD 的定制与开发[M]. 北京:人民邮电出版社,1998.

[5] 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.

[6] 周振红,毕苏萍,张成才. Fortran COM 组件集成到 NET 平台(二)[J]. 武汉大学学报:工学版,2006,39(6):51-54.

Delphi Driving AutoCAD to Draw by Way of Dual Interfaces

ZHAO Feng¹, BI Su-ping², ZHOU Zhen-hong³

(1. Department of Scientific Research, Zhengzhou University, Zhengzhou 450001, China; 2. School of Civil Engineering, Zhengzhou University, Zhengzhou 450001, China; 3. School of Environment and Water Conservancy, Zhengzhou University, Zhengzhou 450001, China)

Abstract: By the use of the programming model of AutoCAD, drawing may be effectively carried out by batch-processing, and applications may possess powerful plotting function. Generally, VB/VBA is used to control AutoCAD to draw by calling the IDispatch interface of its object model, the application having ineffective performance. In this paper, it is put forward how Delphi drives AutoCAD to draw by calling the dual interfaces of AutoCAD's object model, the program of representative application being provided. It has been proved by the experiment that the plotting method both offers programming flexibility and makes the application possess effective performance, thus presenting a new way of driving AutoCAD to plot by the programming.

Key words: Delphi; AutoCAD; programming model; component object model; dual interfaces