

基于 VC++ 的 DXF 数据文件接口的研究

王子茹, 任清波

(大连理工大学土木水利学院 辽宁 大连 116024)

[摘要] 研究了 VC++ 与 CAD 之间数据交换时的接口技术. 根据 DXF 文件格式, 结合 VC++ 技术的图形处理、数据管理功能, 本文开发设计出数据交换接口程序, 可以在 VC++ 程序中实现对 CAD 图形的调用以及二次开发. 实例证明: 该方法方便、快捷、实用.

[关键词] VC++; DXF 文件; 接口技术; 二次开发

[中图分类号] TP 312 **[文献标识码]** A **[文章编号]** 1008 - 3804 (2007) 01 - 0026 - 05

0 引言

随着计算机技术在工程领域的应用, VC++ 的三维可视化开发已经成为当前研究的热点问题之一. 三维可视化开发的数据来源主要有以下三点: 一是通过接口程序导入数据, 这方面 3DS 与 VC++ 的数据接口做得比较成熟^[1]; 二是试验数据, 将试验数据做成文本格式, 然后通过程序调用建立可视化模型; 三是利用 CAD 模型数据. 其中 CAD 模型数据在工程领域应用最为广泛. 因此, 研究 CAD 与各种开发平台之间的数据接口是实现三维可视化的关键. 在这方面, 许多学者做过研究, 文献 [2~4] 对 CAD 与 C 语言的数据接口程序进行了探讨, 文献 [5] 实现了用模板类存储数据的 DXF 文件的接口设计, 文献 [6] 讨论了基于 VC++ 的 DXF 文件的数据提取算法. 本文根据 DXF 文件的数据格式, 对 VC++ 与 DXF 数据文件接口进一步开展研究, 着重解决海量数据图形的读取、保存及输出.

1 DXF 文件基本格式

1.1 DXF 文件

DXF (Drawing Exchange File) (图形交换文件) 是一种 ASCII 文本文件, 它包含对应的 DWG 文件的全部信息, 用它形成图形速度快. 可以与第三方文件有良好的数据互读^[7], 如 3DMax、MaLab 等都可以直接读取 DXF 文件, 但是作为开发软件的 VC++ 却没有为 DXF 文件设立接口函数.

1.2 DXF 文件的数据格式^[8]

DXF 文件本质上由代码及关联值对组成. 代码 (通常称为组码) 表明其后的值的类型. 使用这些组码和值对, 可以将 DXF 文件组织到由记录组成的区域中, 这些记录由组码和数据项目组成. 在 DXF 文件中, 每个组码和值对都各占一行.

每个段都以一个后跟字符串 SECTDN 的组码 0 开始, 其后是组码 2 和表示该段名称的字符串 (例如, HEADER). 每个段都由定义其元素的组码和值组成. 每个段都以一个后跟字符串 ENDSEC 的组码 0 结束.

DXF 文件完整的结构如下:

[收稿日期] 2006 - 12 - 09

[修回日期] 2007 - 02 - 28

[作者简介] 王子茹 (1956 -), 女, 辽宁铁岭人, 教授, 博士, 从事工程可视化辅助设计、工程图学的研究.

HEADER 段: 包含有关图形的基本信息. 它由 AutoCAD 数据库版本号和一些系统变量组成. 每个参数都包含一个变量名称及其关联的值.

CLASSES 段: 包含应用程序定义的类的信息, 这些类的实例出现在数据库的 BLOCKS、ENTITIES 和 OBJECTS 段中. 类定义在类的层次结构中是固定不变的.

TABLES 段: 包含以下符号表的定义: 应用程序标识表、块参照表、标注样式表、图层表、线型表、文字样式表、用户坐标系表、视图表、视口配置表.

BLOCKS 段: 包含构成图形中每个块参照的块定义和图形图元.

ENTITIES 段: 包含图形中的图形对象 (图元), 其中包括块参照 (插入图元).

OBJECTS 段: 包含图形中的非图形对象. 除图元、符号表记录以及符号表以外的所有对象都存储在此段. OBJECTS 段中的条目样例是包含多线样式和组的词典.

THUMBNA L MAGE 段: 包含图形的预览图像数据. 此段为可选.

2 DXF 文件数据接口程序

2.1 DXF 数据接口的设计流程

图 1 是设计的 DXF 数据接口流程图, 具体步骤如下:

(1) 文件的读取. DXF 文件本质上由代码及关联值对组成, 因此, 在整个文件的读取过程中, 是按组进行读取, 如:

```
BOOL dxReadParam ( HDXF hDxf, int& GroupCode, LPTSTR valustr );
```

其中 hDxf 是文件路径指针; GroupCode 是代码; valustr 是代码对应的关联值.

(2) 数据的处理和保存. 此 DXF 文件接口程序主要是应用于处理工程图形, 所以对于读取的数据, 只需要保存有用的信息. 对于文件中所含有图形版本、图形标准设置等的默认设置, 程序中可以忽略, 没有保存.

此接口程序主要用于读取工程 CAD 图形, 用于 VC++ 的可视化程序开发, 利用程序可以直接绘出一般的图形. 另外, 考虑到数据接口的广泛适应性, 程序留了一个数据输出接口, 可以快速地获得 DXF 文件中所含图形的图元信息, 以便根据设计者需要进行数据处理.

(3) 根据读取图元数据信息绘出图形或显示、输出图元信息, 然后关闭文件, 退出程序.

2.2 DXF 数据接口程序的实现

2.2.1 数据的读取

点是构成所有图元的基础, 在 VC++ 中, 默认的点的设置是二维的, 数据类型是正数: Cpoint (int initX, int initY). 而 DXF 文件里点的形式是双精度类型的, 所以本文重新定义了一个点的结构体:

```
typedef struct tag_REALPOINT
{
    double x;
    double y;
    double z;
```

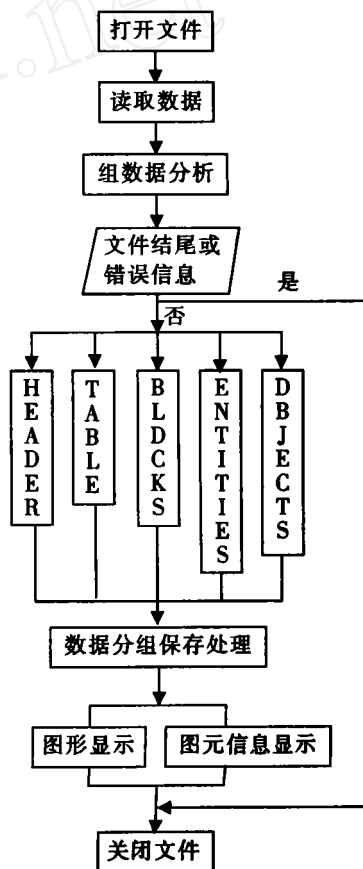


图 1 DXF 数据接口流程图

Fig.1 DXF data interface flow

```
} REALPOINT, * PREALPOINT;
```

这样在读取关于点、线、圆等图元信息时就可以用这个结构体进行存储点的信息了。

如下段程序，DXF文件由代码及关联值对组成，读取时也是按对组值进行读取：

```
char strGroupCode[32];
ReadLine(pDxf, strGroupCode);    //读取一行数据
GroupCode = atoi(strGroupCode);
ReadLine(pDxf, valustr);        //读取一行数据
```

考虑到 DXF数据量比较大，而且有些数据是程序所不需要的，例如 HEADER段，这一段主要包含了与图形有关的系统变量（又称为标题变量）的设置，所以在读取数据的时候，没有对该段进行相应的处理。而 TABLES段，含有大量的关于标注式样、线形、图层、文字式样等信息，BLOCKS段与 ENTITIES段则包含了主要的图元信息，可视化图形信息都在这两个段内，所设计的程序主要针对这三段进行数据读取及处理。以下是一段读取实体段的代码：

```
switch(DxfEntityHeader.EntityType)
{
case ENT_LINE:
    Line.Point0 = ((PDXFENTLINE) &DxfEntityData) ->Point0;
    Line.Point1 = ((PDXFENTLINE) &DxfEntityData) ->Point1;
    dwgAddEntity_Direct(pDrawing, BlockObjhandle, &EntityHeader, &Line);
    break;
case ENT_POINT:
    Point.Point0 = ((PDXFENTPOINT) &DxfEntityData) ->Point0;
    dwgAddEntity_Direct(pDrawing, BlockObjhandle, &EntityHeader, &Point);
    break;
case ENT_CIRCLE:
    Circle.Point0 = ((PDXFENTCIRCLE) &DxfEntityData) ->Point0;
    Circle.Radius = ((PDXFENTCIRCLE) &DxfEntityData) ->Radius;
    dwgAddEntity_Direct(pDrawing, BlockObjhandle, &EntityHeader, &Circle);
    break;
.....
}
```

2.2.2 数据的保存

因为程序在读取如地形图等海量图形信息时，可能需要很大的内存空间，因此，在程序中使用 GlobalLock函数动态的申请一个可移动全局内存块，然后将读取的 DXF文本内容在经过分图元类别处理后，按不同图元类型写入到此内存区域，并根据写入实体的大小，动态调整该内存块的大小。这样就可以像用文本剪贴板一样，随时在不同的地方不同线程调用此内存中的图元信息，而不用反复的保存、读取文本内容。在程序结束时，调用 GlobalFree函数来释放内存块。这样既可以增大读取文件的信息量，又可以加快程序的执行速度。程序如下所示：

```
BOOL AddToEntitiesList(PENTITIES pEntities, PENTITYHEADER pEntityHeader, LPVOID pEntityData,
WORD EntitySize)
{
.....
pEntitiesData = GlobalLock(pEntities->hEntities);
.....
```

```

if(pEntitiesData == NULL)
{
    return FALSE;
}
if((pEntities ->FreePos + sizeof(ENTITYHEADER) + EntitySize) >= pEntities ->TotalSize)
{
    GlobalUnlock(pEntities ->hEntities);
    .....
    pEntities ->hEntities = GlobalReAlloc(pEntities ->hEntities,pEntities ->TotalSize + 65536, GHND );
    .....
}
pEntities指向实体结构 ENTITIES的指针;
pEntityHeader:实体数据头的指针;
pEntityData :将要添加的实体数据结构的指针;
EntitySize :将要添加的实体数据量大小 .

```

2.2.3 图形的绘制

VC++不仅在点定义的时候用的是整数的形式,而且在直线、圆、矩形等绘制的时候所用的参数全是整数类型,这就决定了将导入数据重绘的时候不能直接用 VC++提供的 MoveTo ()、LineTo () 等常用的 GDI 绘图命令. 可以采用两种方法解决这个问题:一是将点的格式转化成像素格式,然后调用 VC++提供的绘图函数,在视口内绘制图形;二是调用 OpenGL 动态连接库,利用 OpenGL 绘图命令进行绘制.

2.2.4 图元信息输出

为了便于直观地了解图形图元信息及二次开发,本程序附带设置了一个图元信息输出接口,使用者可以根据需要对数据进行查询或输出为 Excel 格式文件. 程序中自带了一个 CEditlist 类,负责对图元信息的处理.

3 应用实例

图 2 是 CAD 绘制的地形图. 按上述接口程序进行数据提取及测试,通过运行实现了 CAD 与 VC++ 之间的数据转换,图形输出见图 3,及有关图元信息显示图 (略).



图 2 CAD 等高线地形图
Fig.2 CAD contour terrain map

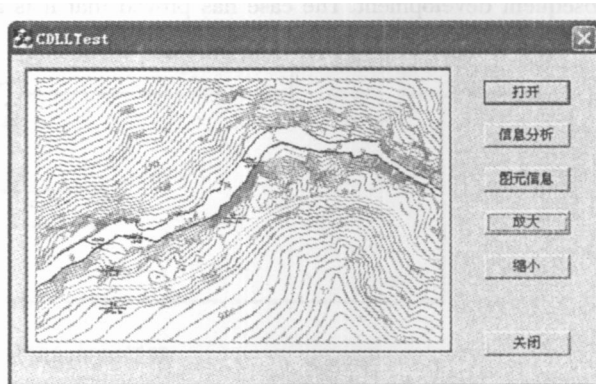


图 3 VC++等高线地形图整体显示
Fig.3 Whole display of VC++ contour terrain map

4 结语

本文研究的 DXF文件数据交换接口程序,是用申请可移动全局内存块保存数据的方法,解决了海量数据读取时容易发生的读取速度慢、内存释放困难的问题。另外,通过设立图元信息的 CSV (EXCEL) 输出,使程序开发者可以轻松的得到 DXF文件中各种图元的数据信息,根据需要对这些图元信息优化处理,进行二次开发。通过实例验证,本接口程序处理像地形图等复杂的 DXF文件,可收到很好的效果。

[参考文献]

- [1] 和平鸽工作室. OpenGL 高级编程与可视化系统开发 [M]. 系统开发篇 (第 2 版). 北京: 中国水利出版社, 2005.
- [2] 张成才, 孙喜梅, 朱陶业. AutoCAD 的 DXF 文件格式及其转换接口研究 [J]. 微型电脑应用, 2001, 17 (8): 54-56
- [3] 张卫峰, 林端敏. AutoCAD 软件的 DXF 接口技术 [J]. 机械与电子, 2004 (11): 78-80
- [4] 景雪琴. VC++ 类和文件技术在 AutoCAD 二次开发中的应用 [J]. 微机发展, 2005, 15 (8): 137-138
- [5] 夏涛, 姜开勇. 基于 DXF 文件的接口设计及两点最短路径的求取 [J]. 仪器仪表用户, 2006, 13 (1): 89-90
- [6] 胡胜红. 使用 VC++ 编程实现 DXF 文件数据提取 [J]. 福建电脑, 2006 11, 189.
- [7] 梁雪春, 宋德明. AutoCAD 二次开发技术指南 [M]. 北京: 清华大学出版, 2001.
- [8] 余承飞, 方勇. AutoCAD2000 二次开发技术 [M]. 北京: 人民邮电出版社, 1999.

Research on DXF Data File Interface Based on VC++

WANG Zi-ru, REN Qing-bo

(School of Civil and Hydraul Engineering, Dalian University of Technology, Dalian 116024, China)

Abstract: This paper studies the DXF interface technology for the VC++ and CAD data exchange. Based on the DXF file formats and combined with VC++ graphic processing and data management technology, a data exchange interface program is developed which realizes transfer of CAD graphics in VC++ program as well as subsequent development. The case has proved that it is a convenient, speedy and practical one.

Key words: VC++ ; DXF file; interface technology; subsequent development