

基于 VC++ 的农田管理地理信息系统(GIS)的开发

孙要夺

(中国农业大学精细农业研究中心)

摘要 虽然现有的 GIS 专业开发工具具有较好的开发平台, 可靠性好、简单易学, 但其存在可扩展性差、对软硬件要求较高、没有系统版权等明显的缺点。结合利用 VC++ 从底层开发应用系统的实践经验, 提出了用 VC++ 从底层开发适合农田管理的基本地理信息系统的设计方法, 利用 VC++ 开发环境在 Windows 98 操作系统下初步实现了农田管理的基本 GIS 的矢量图形系统的开发, 并在此基础上建立了属性数据库, 实现了矢量图形系统与数据库管理系统的双向连接。该系统能够实现基本图形元素的绘制, 并能与其属性数据进行连接, 能够满足农田管理的基本需求。

关键词 GIS; 矢量图形系统; 数据库管理系统

中图分类号 TP 391. 72

Development of Field Management Geographic Information System by VC++

Sun Yaoduo

(Research Center for Precision Agriculture, China Agricultural University, Beijing 100083, China)

Abstract The advantage and the disadvantage of existing professional development tools of GIS (such as the advantage of better developing platform, good security, learning simply and easily, the disadvantage of badly extensible, high need of software and hardware, no system copyright) were analysed. The method of developing field management geographic information system from basis was provided by using VC++ platform. The vector graphics system of basic field management GIS was implemented, the attribute database management system was accomplished, the two-orient connection system between the vector graphics system and the attribute database system were described by VC++ platform in Windows 98 operation system. The system can provide the drawing of basic graphic element, the connection with attribute database, and meet the basic need of the field management.

Key words GIS; vector graphics system; database management system

在 GIS 系统的发展过程中, 已出现了大量的专业开发工具。虽然应用这些专业的开发工具具有许多优点, 如: 较好的开发平台、可靠性好、简单易学; 但也有明显的缺点: 可扩展性差、对软硬件要求较高、没有系统版权^[1]。特别是对于农田管理地理信息系统的开发, 由于农田管理的某些特殊需求, 及基层用户软硬件不易配置且价格昂贵, 这些 GIS 系统专业开发工具的

收稿日期: 2002-03-04

高校博士点基金资助项目(1999001001)

孙要夺, 北京清华东路 17 号 中国农业大学(东校区)63 信箱, 100083

缺点更为突出^[2]。应用VC++从底层开发以信息管理、决策服务及设计为主的实际GIS系统,具有较好的灵活性和可操作性,因此笔者提出了用VC++从底层开发农田管理地理信息系统的开发方法,并利用它实现了系统的开发。

1 相对完整的矢量图形系统的开发

矢量图形系统是GIS系统最重要的组成部分,也是用VC++开发GIS系统的重点所在^[3]。本文中介绍的矢量图形系统具有如下功能:1)相对完善的图形元素的处理功能。具备处理点、直线、圆、连续直线、多边形区域、标注文本等图形元素的能力,并具有处理图例(即图形块)的能力。具有基本图形元素和图例后的矢量图形系统,在很大程度上能够满足普通的管理型GIS系统的矢量图形要求。提供矢量显示图像的功能在很大程度上能够改善矢量图形界面的质量。2)相对完善的图形操作功能。具备图形的放缩、移动、回溯等各种操作功能。3)相对完善的图形输入和输出功能。具备鼠标交互绘制、图形数据交互输入和从打印或绘图设备输出图形的功能。4)具有图层、颜色、线型等的设置功能。5)具有较大的存储容量。处理图形元素的容量能够满足一般实用要求。6)具有较强的容错能力和可恢复性。7)具有较高的处理速度。

在矢量图形系统的开发过程中,最关键的问题是图形元素基类组织的实现。对各种图形元素进行分析,可以发现各类图形元素具有一些相同的属性和操作功能,如图形元素的颜色、线型、线宽、所在层等属性和一个图形元素是否作了删除标志等操作。可以把这些图形元素中共性的东西(属性和操作)组织存放在1个图形元素基类中,具体的图形元素由这个类派生。图形元素基类的具体实现程序如下。

```
class CGraphicDraw: public COBJEKT
{
protected:
    short m_ColorPen;           笔色
    short m_ColorBrush;        填充颜色
    short m_LineWidth;         线宽(像素)
    short m_LineType;          线型
    short m_Layer;             所处图层
    BOOL b_Delete;             是否处于删除状态
    int m_id_only;             图形元素唯一识别号
public:
    CGraphicDraw() {}          构造函数
    CGraphicDraw(short ColorPen, short ColorBrush, short LineWidth,
        short LineType, short Layer, int id_only, BOOL Delete) 构造函数
    {
        m_ColorPen= ColorPen;
        :
    }
}
```

在这个图形元素基类 CGraphicDraw 中, 有 2 个构造函数, 第 1 个构造函数不带参数, 第 2 个带 7 个参数, 用来初始化类中的成员变量。在这个类中, 成员变量 m_id_only 是每个图形元素的唯一识别号。设计完图形元素的基类后, 其他各个具体的图形元素类都从这个类派生。以直线为例

```
class CLine: public CGraphicDraw           直线类
{
protected:
    float m_X1, m_X2, m_Y1, m_Y2;       直线的起点和终点坐标
public:
    CLine() {}                          不带任何参数的构造函数
    以下是有初始化参数的构造参数
    CLine( short ColorPen, short ColorBrush, short LineWid, short LineType, short Layer,
           int id_only, BOOL Delete, float X1, float Y1, float X2, float Y2)
        : CGraphicDraw (ColorPen, ColorBrush, LineWid, LineType, Layer, id_only,
Delete)
    {
        m_X1= X1;
        :
    }
}
```

在这个直线类 CLine 中, 也重载了 2 个构造函数。第 1 个构造函数没有参数, 故可以用不带参数的方式定义一个 CLine 对象; 另一个构造函数有 11 个参数, 并利用前 7 个参数调用基类 CGraphicDraw 的构造函数, 对基类 CGraphicDraw 中的成员变量进行初始化, 在本身的构造函数中对直线类 CLine 自身的 4 个参数(直线的起点和终点)进行初始化。其他基本图形元素类如连续直线或封闭多边形、圆、圆弧、标注文本等的定义和直线类类似, 由于篇幅原因从略。

这部分的难点在于图层显示, 可根据输入文件显示农田、道路、村庄等信息。数字地图采用矢量方式, 可任意放大、缩小、移动, 并能根据用户需要进行地图标注^[4]。

2 数据库管理系统的开发

数据库管理系统是 GIS 系统重要的组成部分, 用来管理 GIS 系统中各种性质的数据。最常用的开发数据库管理系统的方法是使用 ODBC (开放数据库互连)。ODBC 作为一种开放的数据库, 可以使用各种数据库系统 (如 ForPro, SQL Server, Oracle, Access 等) 的数据库文件作为数据源, 使用 SQL 语言作为操作和查询语言, 按照统一的方法实现数据库管理系统。笔者开发的数据库管理系统就是采用这种方法实现的, 且能提供以下几个功能: 1) 数据浏览功能。通过操作界面实现对数据表的浏览。2) 数据编辑功能。能够对数据表中的记录进行增加、修改、删除等操作, 即可以任意修改数据表的内容。3) 查询功能。能够提供操作界面, 任意组织数据表的过滤条件, 对数据表进行查询操作。4) 排序功能。能够组织排序字段, 对数据进行多字段

和可选顺序的排序操作。

为了在矢量图形系统 GraphicDraw 中方便地对图形系统和数据库系统同时进行管理,考虑在应用程序 GraphicDraw 当前视图(CGraphicDrawView 类来管理)之外新建另一个视图,在这个视图中实现对数据记录集合的管理功能。这实际上是一个多视图的操作^[5]。视图的创建过程如下。

- 1) 通过 MFC 的 CView 及其派生类派生一个视图类。
- 2) 新建一个菜单资源。
- 3) 新建一个多文档模板对象,在应用程序类的 OnInitInstance 函数中加入代码,创建一个新的文档模板对象:

```

BOOL CGraphicDraw::OnInitInstance()
{
    ..... 省略以上代码
    DocTemplate1= new CMultiDocTemplate(
        :
        RUNTIME_CLASS(CDatabaseView));      注册视图类
    pDocTemplate= new CMultiDocTemplate(
        :
        RUNTIME_CLASS(CGraphicDrawView));  注册视图类
    AddDocTemplate(pDocTemplate);          增加一个文档模板对象
    ..... 省略以下代码
}

```

- 4) 创建视图。在主框架窗口类 CMainFrame 中修改 OnNewWindow 函数以创建视图,创建完新的视图后,就可以用这个视图来显示数据库的一些操作结果。

3 矢量图形系统与数据库管理系统的连接

在 GIS 系统中,矢量图形系统与数据库管理系统并非相互独立,需要在两者之间建立起连接关系。所谓建立连接关系,就是在矢量图形系统的图形元素与数据库管理系统的数据库记录(或数据库视图)之间建立连接,把性质数据赋给矢量图形元素。

矢量图形系统与数据库管理系统的连接方式有多种,一般采用以下 3 种方式:连接数据表的一条记录,连接整个数据表,连接数据表的一个记录集合。以下介绍的结构可以用来存储一个连接的参数。

```

typedef struct
{
    unsigned char lb;           连接方式: 1—连接数据表的一条记录
                                2—连接数据表的一个记录集合
    unsigned char graph_id;    图形类别号
    int graph_element_id;     图形元素的识别号
    short database_id;        记录集合对象的识别号
}

```

```

int record_id;           在连接方式 1 下, 表示数据记录的唯一识别号
                        在连接方式 2 下, 表示过滤条件的唯一识别号

}LinkStru;
    一个 LinkStru 结构可以用来存储一个图形连接单元和数据连接单元之间连接的完整信息。建立了 LinkStru 结构后, 就可以增加一个与矢量图形元素的连接。
void CLinkDlg::OnLinkAdd()
{
    :
    GetDlgItemText(IDC_GRAPHEDIT, m_str);    从编辑框 IDC_GRAPHEDIT 中
    读出
    m_str.TrimLeft();                        内容到字符串 m_str 中
    去掉字符串 m_str 左边的空格
    if(m_str.GetLength() < 1)               如果字符串长度小于 1, 即没有在
    IDC_GRAPHEDIT 中填入
        return;                             图形类别识别号, 函数退出
    int lb= atoi(m_str);                    得到图形类别识别号
    m_str.Empty();
    :
}
}

```

与增加一个与矢量图形元素的连接类似, linkstru 结构可实现修改、删除与矢量图形元素的连接等操作。最后实现矢量图形系统与数据库管理系统的连接, 实现双向查询, 实现图形元素的空间拓扑关系的信息统计和分析。

这部分的难点在于属性查询: 系统支持对道路、地域和标识物等的多种查询方式, 包括指定任意区域进行查询, 指定图形中实体进行查询, 按地名、行政区划进行查询, 用户在屏幕上指定区域进行查询等方式。查询主要是通过捕捉鼠标位置, 确定鼠标所在位置的地理信息或根据查询条件对整个图形信息进行符合查询条件的搜索查询, 重新显示查询结果。

参 考 文 献

- 1 Shane R, Zhang Naiqian, Randy T. Development of a field-level geographic information system (FIS). In: A S A E Annual International Meeting. Florida: Sponsored by A S A E, 1998 A-15
- 2 汪懋华. “精细农业”发展与工程技术创新. 农业工程学报, 1999, 15(1): 1~ 8
- 3 GPS receiver operation manual. America, Trimble Navigation Limited, 1999. 32~ 52
- 4 马智民. 应用地理信息系统设计与实现. 西安: 西安地图出版社, 1996. 25~ 100
- 5 侯俊杰. 深入浅出 MFC. 武汉: 华中科技大学出版社, 2001. 337~ 397