

# GIS 中地图符号设计系统的设计与实现

程朋根<sup>1),2)</sup> 龚健雅<sup>1)</sup> 眭海刚<sup>1)</sup>

<sup>1)</sup>(武汉测绘科技大学测绘与遥感信息工程国家重点实验室, 武汉 430079)

<sup>2)</sup>(华东地质学院测量系国土与信息工程研究所, 临川 344000)

**摘要** 为了研制国产 GIS 软件——Geostar 地图符号设计子系统, 采用面向对象的方法, 对符号分类组织、图素类设计、符号库结构、符号设计软件及空间信息符号化动态库等进行了设计与软件开发, 所研制的软件成功地对国家标准系列地形图图式进行了设计, 可满足实际生产的需要. 介绍了符号设计子系统设计与开发过程中面向对象方法的应用与技术问题. 研究表明, 采用面向对象的方法来开发地图符号设计软件, 技术思路正确, 所开发的软件具有良好的封装性, 便于维护和扩充.

**关键词** 地理信息系统 面向对象 地图符号 系统设计

中图法分类号: P208 P283.1 P284.1 文献标识码: A 文章编号: 1006-8961(2000)12-1006-06

## Design and Implement of Map Symbol Design System in GIS

CHENG Peng-gen<sup>1),2)</sup>, GONG Jian-ya<sup>1)</sup>, SUI Hai-gang<sup>1)</sup>

<sup>1)</sup>(National Key Lab for Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan Technical University of Surveying and Mapping, Wuhan 430079)

<sup>2)</sup>(Surveying Department, East China Geological Institute, Linchuan 344000)

**Abstract** The research purpose of this paper is to develop map symbol design subsystem for our country's GIS software——Geostar, which is developed by research center of Wuhan Technical University of Surveying and Mapping. Under the technology of object-oriented, symbol classification and organization, graphical element classification design and symbol base structures have been designed, and symbol design software and dynamic link library of spatial information symbolization have been developed. We have designed a series of specifications for cartographic symbols of national standard topographic map by using our map symbol design software, and the results are satisfied to the demand of production. In this paper, we introduce mainly application problems of object-oriented technology in the course of design and developing of symbol design subsystem. The research shows that the technique route we adopted is correct, the developed software has favorable encapsulation and it is convenient to maintenance and expansion.

**Keywords** Geographic information system, Object-oriented, Map symbol, System, Design

## 0 前言

无论是地理信息系统, 还是地图制图系统, 地图符号设计均是它们的主要功能之一, 且空间实体符号化(即地图制图输出)也是地理信息系统(GIS)的

重要内容. 由于 GIS 应用领域的不同, 所需的符号体系也不尽相同, 因此为提高地理信息系统的适应性, 地理信息系统应具有符号设计、地图制图的功能.

综观目前的 CAD 与 GIS 软件, 符号设计的实现途径有如下 4 种: 一是文本编辑器设计的方法, 如

AutoCAD 的图形文件(.shp)、线文件和阴影文件,其特点是设计速度慢,不能实时观察所设计的符号;二是采用系统提供的二次开发语言编程,如 Arc/INFO 的 AML 语言、MGE 的 MDL 等,它提供了编程实现符号绘制的接口;三是利用系统本身的图形编辑功能,如 AutoCAD 的块文件(block)、MEG 的单元(cell),但这种方式又受系统图形编辑功能的限制,且只能设计点符号;四是用所提供符号设计功能,如 MGE 提供了线型编辑(Line Style Editor)功能可用于交互式编辑线符号.总之,它们都有各自的特点和不足,目前业界一致的看法是为系统提供一个符号编辑器(即符号设计系统),以用于符号设计,其对符号设计系统的要求是:(1)组成符号的图素需满足符号设计要求;(2)设计时,能实时观察所设计的符号;(3)符号设计系统界面友好、操作方便灵活;(4)符号设计精度应满足一定的精度要求;(5)符号设计系统、符号显示软件应具有良好的封装性、可维护性和可适应性.解决这一问题的关键是在符号设计系统的符号库组织、程序设计等方面需以面向对象的技术为中心.在所研制的国产 GIS 软件 Geostar 中,其符号设计子系统正是基于这一思想而开发的.

## 1 地图符号特征与符号图素设计

### 1.1 地图符号特征分析

现实世界的事物形态各异、千变万化,但描述现实世界的图件通常采用地形图和各类专题图.一般它们可看作是符号的集合,而且从几何角度看,描述地物的符号不外乎为点符号、线符号、面符号、专题符号.

实际上,点符号、线符号、面符号虽各有其特点,

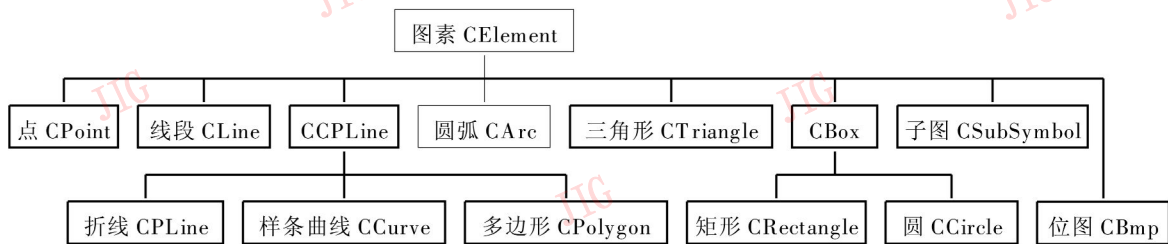


图 1 点符号的图素对象类之间关系图

### 1.2.2 线状符号图素设计

一般线符号绘制采用组合绘制方法,而且任何线符号均可以由具有单一特征的线符号组合而成,例如栅栏符号由虚线、连续点符号、齿线符号 3 种对

但又具有共性,它们的差异仅是构成各自的基本图素不同,而它们的绘制参数(符号代码、绘图句柄、笔的颜色、刷子的颜色等)和操作方法(绘制、删除等)则基本一致.根据面向对象的观点,为使各类符号对象具有相对独立性,需先将点符号、线符号、面符号定义成 3 种符号对象类,并将各类符号的数据成员(属性数据)及其函数成员(操作方法)封装在各自的对象类中,然后在这 3 个对象类的基础上,概括出更高层次的超类,即符号类.

### 1.2 符号图素设计

由于符号是图素的集合,因此图素的设计是设计系统的核心.一般根据点、线、面符号的不同特点,即可归纳设计出组成符号的基本图素,并将各基本图素定义成不同的图素对象类.这样,一个符号则是不同图素对象类实例对象聚集而成的复杂对象,而对具有相同数据的成员和一致操作方法的对象类,则可进一步抽象出更高层次的超类.

#### 1.2.1 点状符号图素设计

由于组成点符号的图素可以分为点、折线、多边形等 11 种,因而又可将组成点符号的图素分成点类、线段类、折线类、样条曲线类、圆弧类、圆类、三角形类、矩形类、多边形类、子图类、位图类等 11 类.且各类图素可根据其属性值的不同而产生不同的图素,如多边形有空心和实心的多边形,空心的多边形又有压盖和不压盖之分,实心的多边形又分实心填充和位图填充.由于矩形和圆在设计时具有相同的定位过程和操作方法,且定位数据成员相同,因此可抽象出一个 CBox 类,同样,对折线、多边形和样条曲线也可以抽象出一个 CCPLine 类.此外,还可进一步将各类图素对象中相同的数据成员和操作方法抽象到更高层的图素超类(CElement)中.这样,组成点符号的图素对象类之间关系即可用图 1 表示.

象聚集而成(如图 2 所示).

由于针对地形图图式线符号,可设计出如下 13 种组成线符号的基本线型:实线、虚线、点虚线、双虚线、双实线、连续点符号、定位点符号、导线连线、

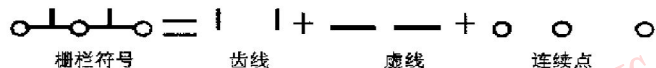


图2 栅栏符号的组成

导线点符号、齿线符号、渐变宽实线、渐变宽虚线、带状晕线,故可将上述基本线型分别设计成组成线符号的13种图素对象类。另外,由于不同的线图素对象有其不同的特征,且描述它们的数据成员也不相同,因此应针对不同的图素对象,来设计数据成员,如定位点符号对象类数据成员为点状符号代码、符号缩放系数、绘端点、指正北等。

### 1.2.3 面状符号图素设计

一般绘制面符号的方法有阴影线填充、点符号填充和位图填充3种,并且对面符号而言,可设计阴影线填充类、点符号填充类、位图填充类3种图素对象类。它们的数据成员如下:

阴影线填充类有倾角、线宽、起始位置( $x, y$ )、偏移量( $dx, dy$ )、实部长、虚部长、(线)色。

点符号填充类有行偏移、列偏移、行间距、列间距、缩放系数、旋转角、点符号、旋转角形式(固定、随机)、布点形式(品字形、井字形)。

位图填充类有位图长度、位图宽度、行间距、列间距、缩放系数、旋转角、填充形式(品字形、井字形)、位图。

## 2 符号库结构设计

### 2.1 设计系统符号库结构设计

设计系统的符号库,即为面向符号设计系统的符号库。因为符号库就是符号描述信息的集合,因此无论是点符号、线符号,还是面符号,都是各自图素对象类实例对象聚集而成的复杂对象。为了节省存储空间,符号库仅保存组成符号的图素对象标识和图素描述参数。但是,对符号库来讲,仅存放符号描述信息的数据文件还不行,还应建立索引机制。其索引机制建立的方法有如下两种:一是索引与数据放在同一文件中,但索引存放在数据文件之前,即符号库分索引区和数据区;二是索引与数据分开存放,即索引存放在一个文件,数据存放在另一文件。这两种索引建立的方法,前者的特点是一个符号库仅为一个文件,从而便于管理,但不便添加符号,这种方式一般用于设计系统输出符号库的组织;后者是一个符号库由两个文件组成,为便于管理,可采用文件名

相同而后缀不同的方法来组织符号库。

为便于操作,本文符号设计系统采用了数据与索引分开存放的方式,而设计系统向应用系统输出的符号库,则采用索引在前,数据在后的方法组织。

由于点、线、面各种符号的符号特征不同,因此索引结构会有所区别,但3种符号索引结构中都应该包括符号名、符号代码、描述数据指针、描述数据大小等项,如点状符号库索引文件记录结构为符号名、符号代码、描述数据指针、描述数据大小、点符号类型、定位方式、界面显示比例尺。

而符号描述数据则是“图素代码+图素描述参数”的集合。

### 2.2 应用系统符号库结构设计

应用系统符号库即为面向应用系统(空间实体符号化动态库)的符号库。为了使GIS系统能够管理多比例尺系列、多种形式的符号库,应采用一定的机制予以实现,即可以设计一个符号库索引文件用于符号库的管理,且索引文件的一条记录对应一个符号库,那么符号设计系统每生成一个新符号库就在符号库索引文件中增加一条记录。该符号库索引文件的记录结构为符号库序号、符号库类型(点、线、面、程序界面)、符号库名、符号库描述信息。

这里,引入符号库序号的目的是为了快速获得某个符号的描述信息,因为动态库中是采用“符号库序号 $\times$ 1000000+符号在符号库中的序号”来索引符号的,而不是采用符号代码,这样在绘制符号时,只需查找两次,就可以得到符号描述信息,因而极大地提高了获取符号描述信息的速度。而且建立了这样的符号库索引文件之后,用户就可以设计不同比例尺、不同类型的符号库,从而为GIS支持多比例尺制图输出提供了极大的方便。

## 3 符号设计系统设计

### 3.1 功能设计

一般对符号设计系统的要求是系统能实现点符号设计、线符号设计、面符号设计等功能。由于点、线、面符号特征的差异性,因而符号设计界面采取了变通的方法,虽不要求严格一致,但符号设计时的“实时显示”却是符号设计的基础。

该系统采用参数化与图形界面相结合的思想进行组织,其中,点符号采用全图形方式进行设计;线符号、面符号则采用参数化形式,并实时显示所设计

的图形. 该符号设计系统具有编辑、修改、存储、删除、浏览等所有设计符号的功能, 并提供了并入系统库的功能. 这些功能主要包括符号库、符号编辑、参数设置、帮助.

**符号库:** 其主要功能是对符号库文件进行操作, 包括建立一个新符号库、打开一个已存在的符号库、关闭符号库、浏览符号库中的符号、清理符号库中的被删除的符号、将所设计的符号库并入到系统符号库中、合并两个同类型的符号库、退出符号设计.

**符号编辑:** 其主要功能是针对一个符号所进行的操作, 包括创建一个新符号、放弃正在设计的符号、将设计的符号存盘、删除符号库中的符号、选择符号库的符号(点符号设计下, 用于组成新的符号), 以及重绘符号设计窗口, 但对点符号设计而言, 应具有对图素进行删除、移动、旋转、对齐以及前后盖盖处理的功能.

**参数设置:** 主要功能包括在符号设计时, 为提高符号设计精度而设置的改变设计单位、显示或关闭网格点等操作.

其中面符号设计时的符号编辑界面见图 3.

类, 点、线和面符号类与符号浏览类之间则采用类继承的方法来实现. 图 4 为点符号设计模块对象类之间的联系图.

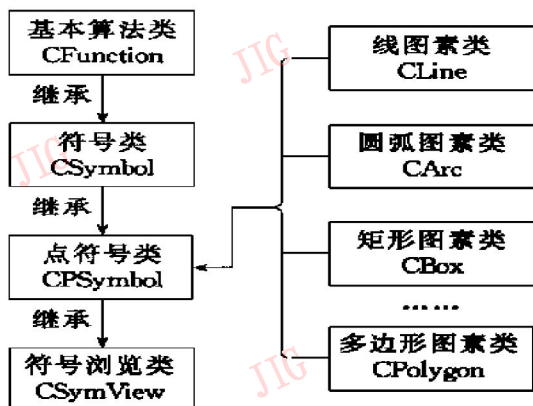


图 4 点符号设计模块对象类之间的联系图

### 4 空间实体符号化动态库设计

由于符号设计的目的是为 GIS 系统建立符号库, 因此为了使 GIS 能调用所设计的符号库, 并实施符号化, 应提供一个空间实体符号化动态库. GIS 中空间实体符号化的过程是, 首先为各类地物配置符号, 然后根据空间实体的空间位置、符号描述信息等参数进行符号化. 由于符号设计系统设计时采用了面向对象的方法, 因而其程序具有良好的封装性和可重用性, 而且动态库设计时, 可充分利用原符号设计系统的源代码, 且原符号设计系统中各对象类之间的关系保持不变, 同时动态库应提供对符号库进行浏览、选择、绘制等接口功能. 为实现这些功能, 还应设计一些新的对象类, 如点、线、面符号选择对话框类, 而动态库中各对象类之间的关系如图 5 所示.



图 3 面符号设计界面

### 3.2 程序设计

程序设计是采用面向对象的方法来进行, 而对于符号来讲, 可将不同图素设计成各种独立的类, 并设计了基本算法类、符号类、点符号类、线符号类、面符号类、符号浏览类; 而对于各种窗口、对话框, 也可以类的形式进行设计. 由于类实现了信息封装, 并具有继承特性, 因而便于程序设计与维护. 而且类之间的联系可以采用类继承和实例对象的方法来实现. 也就是说, 程序设计时, 图素与点符号类、线符号类和面符号类的联系采用实例对象的方法来实现; 而基本算法类与各图素类, 符号类与点、线和面符号

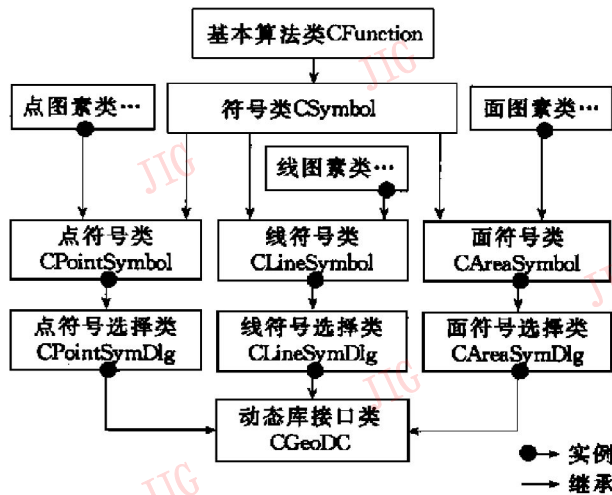


图 5 动态库中各对象类之间的联系图

本文系统动态库接口设计为 CGeoDC 类, 以便于 GIS 其它模块调用. 其 CGeoDC 类的部分定义如下:

```
class AFX_EXT_CLASS CGeoDC: public CDC
{
    DECLARE_DYNAMIC(CGeoDC)
public:
    // 构造/析构函数
    CGeoDC(CWnd * pWnd, BOOL bONPAINT = FALSE); ~CGeoDC();

public:
    // 数据成员
    HWND m_hWnd; // 窗口句柄
    UINT m_currentIndex; // 当前的符号索引值
    double m_currentUnit; // 绘图单位比例系数
    BOOL m_bSymbol; // 是否符号化
    UINT m_nPenWidth; // 当前笔的宽度
    COLORREF m_nPenColor; // 当前笔的颜色
    COLORREF m_currentColor; // 当前前景色
    COLORREF m_nBkColor; // 当前背景色

public: // 函数成员
    int selPointSym(); // 选择点符号
    int selLineSym(); // 选择线符号
    int selAreaSym(); // 选择面符号
    void SymbolPoint(double x, double y,
        UINT Index, COLORREF color,
        double Unit, double scale= 1.0f,
        double angle= 0.0f); // 绘点符号
    void SymbolLine(double * x, double * y,
        int num, UINT Index, COLORREF
        color, double Unit); // 绘线符号
    void SymbolArea(double * x, double * y,
        int num, UINT Index, COLORREF
        color, double Unit); // 绘面符号
    BOOL CreateSymbolBrush(long Index,
        CBrush * abr); // 创建面符号刷子
```

```
# ifdef _DEBUG
virtual void AssertValid() const;
virtual void Dump(CDumpContext& dc) const;
#endif;
```

在该接口类中, 不仅提供了选择和绘制点、线、面符号的公有函数, 而且在这些函数中, 还通过实例对象来建立它们与符号选择(对话框)类、符号类之间的联系. 例如, 选择线符号函数的部分定义如下:

```
int CGeoDC::selLineSym()
{ CLineSymDlg cdlg;
```

```
// 定义线符号选择类实例对象
cdlg.DlgCaption.LoadString(IDS_SYMBOL_LINE);
cdlg.selfiletype= 2;
// 设置线符号类型
if (cdlg.DoModal() != IDOK) return -1;
int m_pFN= cdlg.GetFileIndexNumer();
int m_pSN= cdlg.GetSymbolNumber();
return (m_pFN* 1000000+ m_pSN);
// 返回符号索引值}
```



图6 GeoStar 环境下调用动态库实例

图6是在 GeoStar 环境下为空间实体(地物)配置符号时, 调用空间实体符号化动态库中的线符号所选择函数的结果, 其背景为利用动态库对空间实体进行符号化后的结果.

## 5 结束语

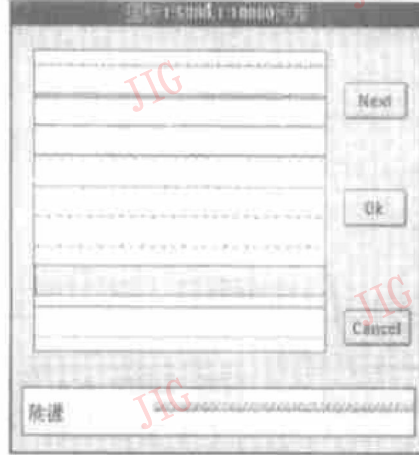
笔者根据本文的技术思路对 GIS 中的地图符号设计子系统进行了设计与开发. 结果表明, 采用面向对象技术开发的地图符号设计软件, 逻辑思路正确、开发方便, 所开发的软件具有良好的封装性, 且便于维护和扩充. 笔者还利用所开发的软件对国家标准系列地形图图式进行了设计, 并建立了相应的符号库, 已可满足实际生产的需要. 这些图式包括:

- (1) 国标 1: 500, 1: 1 000, 1: 2 000 地形图图式;
- (2) 国标 1: 1 000, 1: 10 000 地形图图式;
- (3) 国标 1: 50 000, 1: 100 000 地形图图式;
- (4) 国标 1: 250 000 地形图图式;
- (5) 广州市 1: 500, 1: 1000, 1: 2000 地形图图式.

图7为系统所设计的国标 1: 10 000 万符号库实例.



(a) 点符号库



(b) 线符号库



(c) 面符号库

图 7 国标 1: 10 000 符号库实例

### 参 考 文 献

- 1 陈顺清. 地图符号库设计. 测绘学报, 1993, 22(1): 56~ 63.
- 2 温洪杰等. 图符生成编辑软件. 计算机辅助设计与图形学报, 1992, 4(1): 41~ 45.
- 3 徐庆荣等. 计算机地图制图原理. 武汉: 武汉测绘科技大学出版社, 1993.
- 4 唐荣锡等. 计算机图形学教程. 北京: 科学出版社, 1990.
- 5 古新生等. 面向对象方法与 C++ 新版本. 西安: 西安交通大学出版社, 1992.
- 6 CHENG Peng-gen. Design and organization of vector-based symbol base in GIS[A]. In: Archives of ISPRS Commission, Wuhan: Wuhan Technical University Of Surveying and Mapping, 1994, 30(6): 23~ 28.
- 7 Frank A U, Engenhofer M J. Computer cartography for GIS: An object-oriented view on display transformation. Computer & Geosciences, 1992, 18(8): 975~ 987.



**龚健雅** 1957 年生, 1992 年获武汉测绘科技大学摄影测量与遥感专业博士学位, 现为武汉测绘科技大学测绘遥感信息工程国家重点实验室常务副主任, 教授, 博士生导师, 长江学者奖励计划特聘教授. 目前主要研究领域为地理信息系统理论与关键技术、软件开发、遥感图象处理、空间数据模型与数据结构等.



**眭海刚** 1973 年生, 武汉测绘科技大学摄影测量与遥感专业博士生. 目前主要研究领域为地理信息系统与遥感集成的理论与应用研究等.



**程朋根** 1964 年生, 1996 年获武汉测绘科技大学摄影测量与遥感专业硕士学位, 现为华东地质学院国土与信息工程研究所副所长, 副教授, 武汉测绘科技大学摄影测量与遥感专业在职博士生. 目前主要研究领域为计算机地图制图、地理信息系统理论与软件开发等.