

基于 MapX 的多比例尺地图数据无缝显示

张海堂¹, 罗睿²

(1. 西南电子电信技术研究所, 四川 成都 610041; 2. 总参 61 所, 北京 100031)

Seamless Display of Multiscale Map Data Based on MapX

ZHANG Hai-tang, LUO Rui

摘要: MapX 是 MapInfo 公司推出的一个快速、易用的开发 GIS 系统的 ActiveX 控件。多比例尺地图数据无缝显示是 GIS 系统的必备功能。在分析 MapX 在多比例尺地图显示方面的不足的基础上, 提出一种基于 MapX 建立虚拟工作区的方法, 用以对多比例尺地图数据进行管理和无缝显示, 并给出了在 VC++ 中的具体实现方法。

关键词: MapX; 虚拟工作区; 多比例尺

一、引言

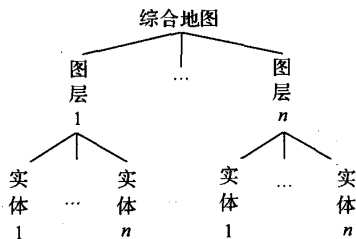
MapX 是 MapInfo 公司推出的一个用于开发 GIS 系统的性价比好, 功能强大的 ActiveX 控件产品。由于它是一种基于 Windows 操作系统的标准组件, 因而能支持绝大多数标准的可视化开发环境如 Visual C++, Visual Basic, Delphi, PowerBuilder 等。利用 MapX, 编程人员在开发过程中可以选用自己最熟悉的开发语言, 轻松地自己的应用中嵌入地图功能。

多比例尺地图数据无缝显示是 GIS 系统必备的功能, 也是 GIS 开发者长期研究的题目之一。本文首先介绍了 MapX 的空间数据组织, 分析了 MapX 在多比例尺地图显示方面的不足, 提出了一种通过建立虚拟工作区的方法, 用以对多比例尺地图数据的管理和无缝显示。

二、MapX 的数据组织与地图显示

1. MapX 的空间数据组织

空间数据组织是 GIS 系统的基础。MapX 的空间数据组织如图 1。



MapX 是按地图分层的方式进行空间数据组织。用户可以通过图形分层技术, 根据自己的需求或一定的标准对各种空间实体进行分层组合, 将一张地图分成不同图层。在进行地图显示时, 采用这种分层存放的结构, 可以提高图形的搜索速度, 便于各种不同数据的灵活调用、更新、管理和显示。

地图图层是 MapX 空间数据组织的基本存储单位。在图层内, MapX 采取的是一种基于空间实体和空间索引相结合的空间数据结构。空间实体是地理图形的抽象模型, 主要包括点、线、面 3 种类型。任何点、线、面实体的图形信息都可以用直角坐标点 x, y 来表示。点可以表示成一对坐标 (x, y) , 对于线和面, 则均被表示成由多个坐标对构成的集合 $\{(x_i, y_i), i = 1, 2, \dots, n\}$, 其中 n 为坐标对个数。空间索引是对空间实体的一种管理机制。通过空间索引, 就能够在 MapX 中以尽量快的速度查询到给定坐标范围内的空间实体及其所对应的数据, 从而提高空间操作的速度和效率。在进行图层数据存储时, MapX 采用的存储方式是图形数据、属性数据和索引数据分开存储。每个图层数据存储为多个文件: ① 属性数据的表结构文件. TAB; ② 属性数据文件. DAT; ③ 交叉索引文件. ID; ④ 空间数据文件. MAP; ⑤ 索引文件. IND。

2. MapX 的主要模型结构

MapX 是基于面向对象语言开发的组件, 其基本组成单元是 Object (单个对象) 和 Collection (集合)。其中对象具有属性和方法, 集合则是多个对象的组合。每种对象和集合负责处理地图某一方面的功能。

如图 2, MapX 顶层是 Map 对象本身, Layers, Datasets, Annotations 是 Map 对象的 3 个重要的组成。其中, Layers 为图层对象 Layer 的集合, 主要用于操作地图的图层。每一个 Layer 对象包含由多个实体对象 Feature 组成的集合 Features。Datasets 为数据集对象 Dataset 的集合, 用于访问空间数据表。Annotation 则用于在地图上增加文本或者符号。另外, MapX 的其他对象如 Field, Style 等主要用于对以上主要对象的描述、管理和操作等。

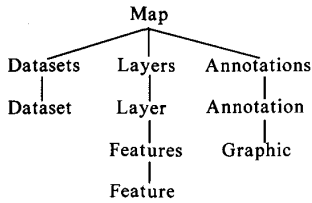


图 2 MapX 主要模型结构

3. 利用 MapX 进行地图显示

以 VC++ 6.0 为例, 在用 MapX 进行地图显示时, 需要先在显示的窗口类内, 以 CMap View 为例, 创建一个 Map 对象。

```

Int CMapView:: OnCreate ( LPCREATE
STRUCT lpCreateStruct)
{
.....
CMapX m_ ctrlMapX;
m_ ctrlMapX. Create(NULL, S_ VISIBLE,
CRect(0,0,100,100), this,
IDC_ MAP1);
.....
}
  
```

在显示窗口类的 WM_ SIZE 事件中设置地图显示窗口的范围:

```

void CMapView:: OnSize( UINT nType, int cx,
int cy)
{
CView:: OnSize(nType, cx, cy);
if (cx != 0 && cy != 0)
m_ ctrlMapX. MoveWindow(0,0,cx,cy,
TRUE);
}
  
```

在 MapX 中, 根据不同的专题属性将地图分成不同的层。由于显示的地图往往不是只针对某一个图层, 所以可以通过使用 MapX 的地图管理工具 Geoset Manager, 将需要的所有的图层表文件集中

在一起, 连同每一层的设置一并存储到一个 . GST 文件中。在进行地图显示时, 需要为 Map 对象设置一个地图数据集:

```
m_ ctrlMapX. SetGeoSet(" * . gst");
```

最后通过 void CMapX:: SetCurrentTool (long) 函数为 m_ ctrlMapX 设置地图操作参数就可以进行地图缩放、漫游等操作了。

4. 利用 MapX 进行多比例尺地图显示的不足

在利用 MapX 进行地图显示时, 一次只能显示一个 Geoset, 因此需要事先将需要显示的地图的所有图层加入一个 Geoset 的数据集中。由于 MapX 没有记录地图的比例尺参数, 在作多比例尺地图显示时, 需要人工为每一个图层设置固定的显示参数, 如图层的最大最小显示尺度。要实现直接利用 MapX 进行多图幅漫游, 需先利用 MapInfo 软件工具将多幅图拼接成一幅图, 因此对于大范围内多幅多比例尺地图数据显示与无缝漫游来说, 不仅人工操作复杂, 而且数据量大, 放在一个 Geoset 中, 显示速度较慢。

三、基于 MapX 的虚拟工作区的多比例尺地图显示

1. 基于 MapX 的虚拟工作区的建立

MapX 控件通过对象 CMapX 来进行地图的显示、操作和管理。CMapX 相当于一个地图工作区, 在其上实现 GIS 的所有功能。虚拟工作区是与 CMapX 协同工作的工作区, 用以实现多比例尺地图的显示与漫游。多比例尺地图显示与漫游涉及不同图幅以及不同比例尺数据, 每一个图幅可用一个 Geoset 进行管理。虚拟工作区的属性与方法定义如下:

```

class CVirtualMapX : public CObject{
//Attributes:
CMapX * m_ pCtrlMapX;
CString m_ strDataSourceName; //数据源名称
CString m_ strDatasetSourceDesc; //数据源路径
CArray < CString, CString& > m_ strGeosetPaths; //图幅路径
CArray < CString, CString& > m_ strGeosetName; //图幅名称
CMapXRectangle m_ Bounds; //所有数据的边界矩形
CMapXRectangle m_ LoadedBounds; //所有
  
```

