

## 第五章 智能化要素

本章内容：

- 要素的性质
- 智能化要素的步骤
- geodatabase 设计
- 在表格中存储数据
- 要素的形状和范围
- 属性：对象的性质
- 使用子类添加行为
- 属性验证
- 对象之间的关联

现实世界中的所有事物都遵循一定的自然规则和关联规则。水是往低处流的，道路要依据交通等级归类，宗地分界总与契约对应。使用 ArcInfo 中子类的属性验证规则，我们就可以对事物的行为进行更充分的表达。

## 5.1 要素的性质

现实世界中的地理对象的联系是极其丰富的。每一个事物都与周边对象有其特定的联系。它们占据一个位置、拥有一个轮廓或者一片区域，同时还拥有邻接的对象——每一个地理对象，都会扮演资源消费者或资源供给者的角色，和其环境周围其它事物交互。同时，事物具有诸如价值、数目、种类或描述等属性信息，还会对应外部环境的演变做出可预期的反应。

### 5.1.1 Geodatabase 数据模型中的要素

就像在 geodatabase 数据模型中描述的那样，要素有许多性质如形状、关联、属性以及行为等。这些性质共同地表达了地理对象丰富的关联背景。

在许多应用中，矢量要素是最常用的表达地理数据的数据格式。它适合于表达离散的有明显边界且稳定存在的地理对象。对于其它作为连续现象存在的地理数据，最好用栅格数据或 TIN 来建模。

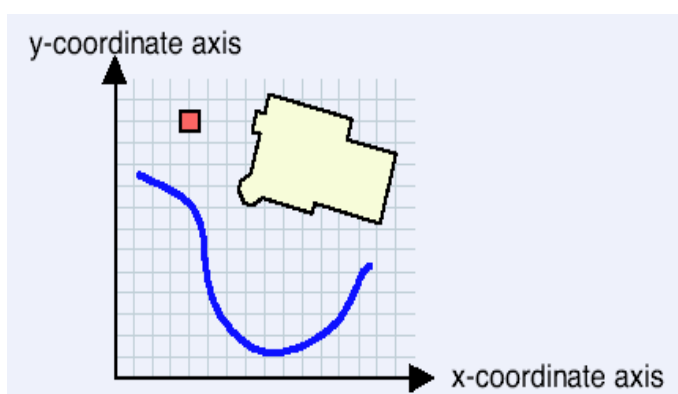
本章讨论如何将 geodatabase 中的矢量数据“智能化”。下面做一个简要的认识。

#### 5.1.1.1、要素具有形状

要素的形状是以 geometry 这么一个特殊字段存储在要素类表中的。要素可以用以下这些几何类型表达：

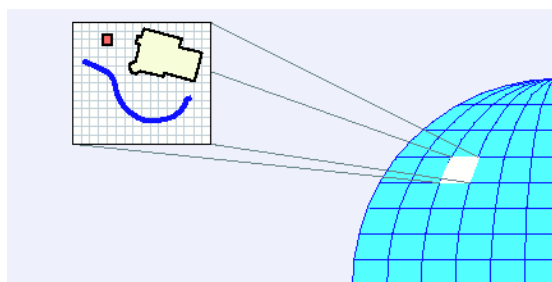
- 点或多点（一组点）
- 线（一组相连或不相连的线段）
- 多边形（不相邻或嵌套的圆环）。圆环是由一组连接的、闭合的、不相交的线段组成的

组成线和多边形的线段可以是直的线段、圆弧、贝塞尔（Bezier）曲线和椭圆弧。



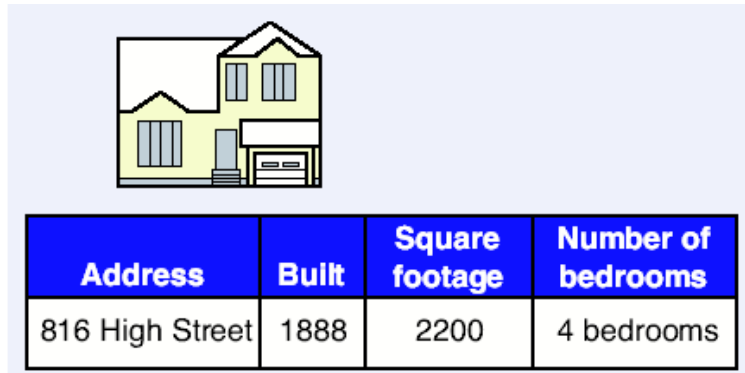
#### 5.1.1.2、要素具有空间参考

在迪卡尔坐标系统中要素的形状以 x、y 值存储。但我们知道，地球表面是粗略的椭球体。空间参考详尽地描述了如何将一组要素的 x、y 坐标映射到地球表面上去。



#### 5.1.1.3、要素具有属性

要素以要素类表中的字段来保存属性。要素类表是关系数据库的表格。属性定义了要素标准的常规特性，这些特性可以用数值、文本或其它描述信息表示。



Address	Built	Square footage	Number of bedrooms
816 High Street	1888	2200	4 bedrooms

#### 5.1.1.4 要素具有子类

要素的集合便成为要素类。要素类是一组同类要素的集合，当然要素类中，要素之间肯定会存在一定的差异。

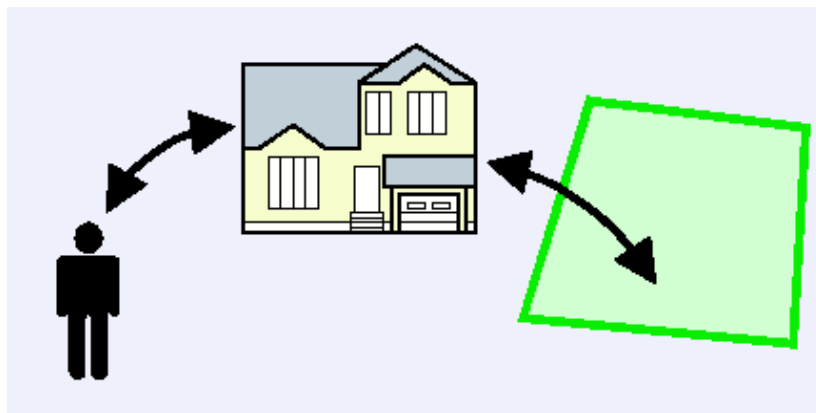
由建筑物组成的要素类可以被逻辑地分成居民地、商业用地和工业用地等子类。使用子类，你可以细化控制要素的属性，如属性域或属性规则等。



#### 5.1.1.5 要素具有关联

所有地理对象都与其它对象存在关联。你可以在不同的要素类中定义地理对象之间的关联。

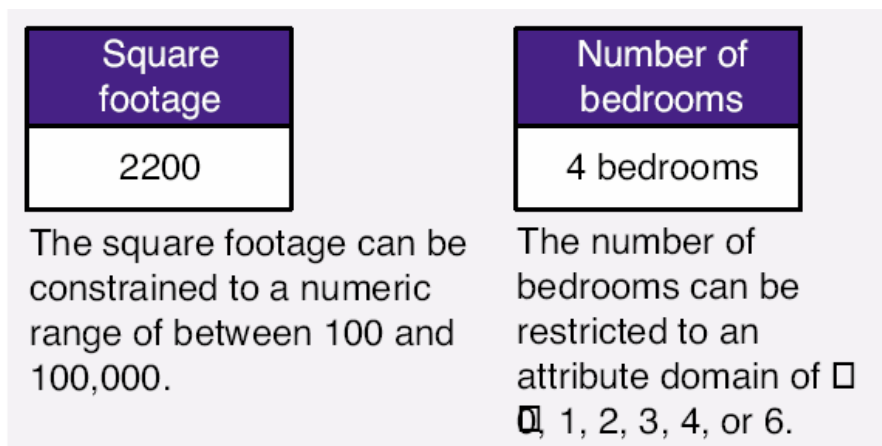
你也可以定义非空间对象的关联，如房屋与其主人的关系。



#### 5.1.1.6 要素属性可以被限制

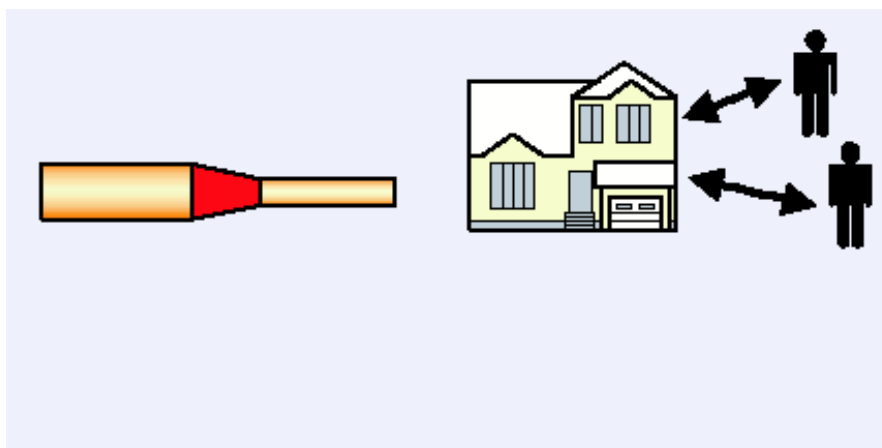
为加强数据录入的准确性，还可以制定属性域对要素的属性进行限定。属性域，表现为一个数值范围或合法值的列表，也可以在要素创建之时为其属性自动分配一个缺省值。你可

以在要素类中为不同的子类设置不同的属性域和缺省值。



#### 5.1.1.7、要素能用规则来验证

现实世界中的对象存在或改变都是必须遵循一定规则的。你可以用这样的规则来限制几何网络中元素的制约规则，或者定义这些元素的关联的对应基数。

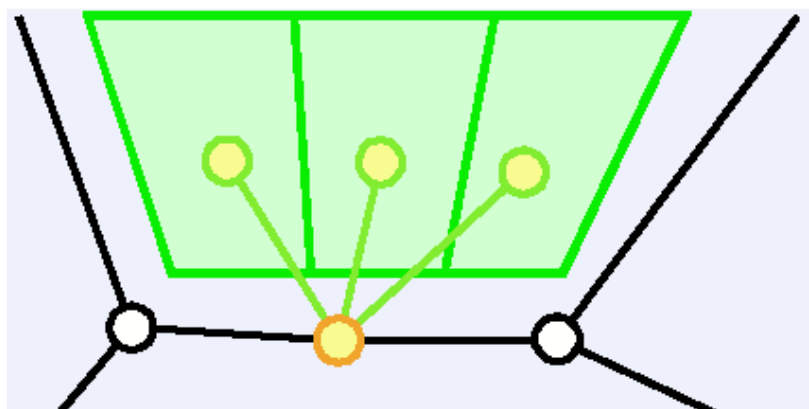


#### 5.1.1.8、要素具有拓扑关系

各类型要素之间具有精确的空间位置关系就叫做拓扑。

例如，宗地的二级小分块必须是彼此严格毗邻的，不允许有缝隙和重叠。这种二维关系称为平面拓扑。

几何网络中的线和应用设施必须是不间断连接的，并且这种连接关系必须明确定义。这样的一维关系图可称为几何网络。

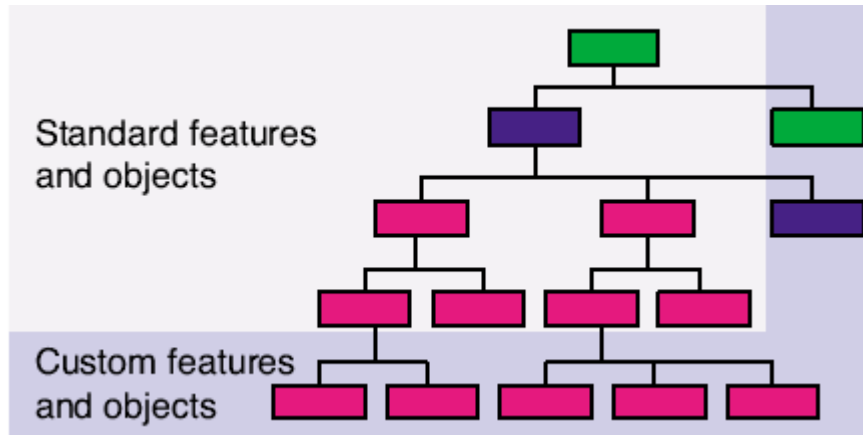


#### 5.1.1.9、要素具有复杂的行为

要素的简单行为是通过选择要素类型和拓扑关系来实现的，并且建立关联、指定属性域

以及属性验证规则等。

更多复杂的要素行为的实现是通过扩展标准要素和为自定义要素编写代码。自定义要素允许复杂的行为，如定制编辑交互、内置分析功能和复杂的图形符号化。



### 5.1.2 智能化要素

Geodatabase 中的要素具有这样的一个框架。在这个框架中，要素具有属性、几何形状、空间参考、关联、域、验证规则、拓扑关系和定制的其他对象属性等。在这一框架中，除了复杂行为都不需要编写程序，geodatabase 已经实现了足以满足很多应用需求的功能。

使用 Geodatabase 要素，你会获得相当丰富的表现手法，得以更自然的表现自然世界。

## 5.2 智能化要素的步骤

设计和创建 geodatabase 时，需要循序渐进的给要素添加“智慧”。可以有选择的进行要素的功能选择。

依据实际应用的需求及其复杂程度，有可能只需要上述那些步骤方法的一部分。例如，大部分实践应用不需要自定义对象。另外一些应用中，并不需求建立要素和对象之间的关联，拓扑关系也不一定是每个要素类都需要的。

### 5.2.1 逐步添加“智慧”

接下来我们将看到定制对象方法的详细步骤。

#### 5.2.1.1、选择要素类型和拓扑关系

在实施数据建模之前，你应该制作一个详细的清单，用于记录你的 geodatabase 中需要创建的所有的对象类型。从清单中，你可以将带有空间参考、拓扑关系和相似制图内容的要素类组织到要素集中。

另外，对于不同的对象采用不同的处理方式。对于非空间对象，创建对象类。而对于空间对象，用点、线和面来创建简单要素类。对于拓扑要素，在要素集中，创建拓扑要素类的列表。

#### 5.2.1.2、设置属性和子类

完成对象或要素类的类型的定义后，便可以为你的对象添加属性字段了。

对象和要素可以包含“子类”(subtype)这个特殊的属性字段。子类用于将对象进行逻辑再分组，可以利用子类字段来表达相似对象(或要素)组之间的差异。需要指出，这样的子类并不需要添加新的对象或要素类，只是在逻辑上把它们进行了归类。

比如道路要素中，使用子类可以将道路分为泥路、居住区道路和高速公路等。子类通过

属性域、缺省值、连通规则和关联规则来增强数据的完整性。

#### 5.2.1.3、定义属性域和验证规则

属性域，是一个指定的合法的属性值集合或范围。使用属性域可以避免在属性赋值的时候出现操作失误。

缺省值为每一个新创建的对象赋予预先设定的默认属性值。这样，可以对要素进行批量的赋值，减少数据录入时间。

连通规则用于网络中的要素，这些规则用以检验某一类型要素是否与另一类型要素正确相连通。

#### 5.2.1.4、建立对象关联

对象之间存在相互作用。对象之间不能通过拓扑关系获取的联系可以使用关联来实现。关联存储在关联类（Relationship class）中，在关联类中可以控制、定义对象（要素）的创建、修改和删除等。

你可以在关联类中定义关联规则，进一步定义要素或对象之间的关联关系。

#### 5.2.1.5、创建定制对象

对象类、域、缺省值、验证规则和关联可以表达一个对象所需的大部分行为。当然，有时可能需要更多的复杂行为如绘制、编辑或检查对象，这需要另外编写代码。ArcInfo 的对象和要素类集合可以由程序员编写代码扩展，以创建得到复杂且高度专业化的对象和要素。

### 5.2.2 总结

在 ArcInfo 中，用户对要素和对象的大部分需求都可以由 geodatabase 这个数据模型来完成，并不需要额外编写程序代码。

作为一个数据建模者，最主要的目标是在数据模型框架内获取对象尽可能多的自然行为。自定义对象和要素及编写代码只有在高级的专业化应用中才会使用。

## 5.3 Geodatabases 设计

地理要素存在于现实世界中，与周围环境存在丰富联系，包括拓扑关系、空间参考和关联等等，在 geodatabase 设计的时候，建模者是有许多方案的。

以下说明 geodatabase 设计过程中需要注意的事项。

### 5.3.1 创建 geodatabase

你可以在 ArcInfo 中创建多个 geodatabase。通常，根据应用的不同，将要素集组合或分离组织到 geodatabase 中，效果会更好。

如果遇到以下情况，需要将要素组合组织到同一个 geodatabase 中：

- 如果一组对象或要素具有关联，那么它们必须存放在同一个 geodatabase 中。
- 具有拓扑关系的要素必须存放在同一个 geodatabase 的同一个要素集中。
- 如果你要并发编辑一组要素，它们必须存放在同一 geodatabase 中。你可以在 ArcMap 中同时浏览多个 geodatabase，但是一次只能编辑一个 geodatabase。

以下是将要素分离组织到不同 geodatabase 中的一些原因：

- 企业拥有很多部门，拥有不同的数据，并且这些数据有不同部门负责，这时可以将要素类分离组织到不同的 Geodatabase 中。
- 如果你操作的是 personal geodatabase，因为有规模限制，所以需要为 geodatabase 做专题或空间上的相应分割。

### 5.3.2 要素集 ( feature dataset ) 和要素类 ( feature class )

Geodatabase 包含三种基本类型的类 :对象( object ) 要素( feature )和关联( relationship )。在 geodatabase 中, 这些类可以按要素集存储或作为独立的类存储。以下情况中, 可以将要素类存放到要素集中 :

- 如果要素类是通过几何网络或平面拓扑而拓扑关联的, 那么必须组织到同一要素集中
- 如果你需要强调一组要素类必须具备共同的空间参考, 那么将这些要素类存放到同一要素集中
- 在一个要素集中, 可以进行任意专题分组关联的类

对关联类的设置没有特别的限制, 关联类可以存放在 geodatabase 中的任意位置, 并能表现整个 geodatabase 中的源类和目标类。如果关联类的源类和目标类存放在同一要素集中, 那是最好不过的, 但这个条件不是必须的。

### 5.3.3 应用子类

有时候, 对于一组要素类, 你有两个选择: 一是将这组要素类作为一个要素类组织, 然后使用子类将这组要素类中性质相近的要素进行逻辑分组; 二是将这组要素分离到不同的要素类中存储, 进行物理分组。

子类是对要素 ( 或对象 ) 类的再进一步分类。使用子类的目的是提高操作性能。在同一个 geodatabase 中, 如果只有一两个要素类, 那么操作速度会比一二十个要素类的操作快捷。

子类让你控制要素类中分组要素的特定的行为, 通过属性规则、缺省值、连通性规则和关联规则等来实现。在大多数情况下, 建议使用子类来将庞大的相关联要素分类。

以下情况中, 不能够使用子类而是使用多个要素类来组织要素 :

- 相关联的要素的不同分组, 需要不同的自定义行为
- 相关联要素, 属性字段方案不同 ( 一个要素类中所有要素必须具有相同属性集合 )
- 需要对相关联要素中的每组要素设置不同的访问权限
- 需要对相关联要素中的某些要素进行版本访问, 而另一些不需要进行访问

作者 : Michael Zeiler

译 : 邢超、孙朝阳

校 : 邢超

声明 : 本书版权系 ESRI 出版社所有。本书是 ArcGIS 学习精品, 不做于商业用途, 属个人爱好, 与广大 GIS 爱好者共同分享。