

## 1.5 地理数据库服务

有了 ArcSDE 软件,这个空间数据引擎提供, ArcInfo 便能够访问空间数据库。在 ArcSDE 中,你可以创建自己的 Geodatabase,并且,这个 Geodatabase 在理论上可以是无限大的!同时, ArcSDE 还提供关系型数据库的开放接口。

### 1.5.1 Geodatabase 如何扩展数据库

Geodatabase 是表达地理信息的数据模型,在很多方面增强了关系数据库的技术:

- Geodatabase 可以将地理数据描述为四种形式:不连续的对象描述为矢量数据,连续对象描述为栅格数据,表面描述为 TINs,将位址描述成为 Locatro 或 address。
- Geodatabase 存储要素的几何形状, ArcInfo 中提供空间操作功能,比如查找要素的邻近、相连或相交的对象。Geodatabase 中还提供定义和管理数据的地理坐标系统的框架。
- Geodatabase 中的拓扑关系可以模拟道路运输事业或其它公用设施网络,进行地理操作,线性要素的拓扑功能是在几何网络 ( Geometric Network ) 中完成的
- ArcGIS Desktop 8.3 出现后, Geodatabase 中的拓扑工具 ( Topology ) 还可以定义不同类型的要素之间的拓扑关系。
- Geodatabase 中可以定义对象、要素之间的关联 ( Relationships )
- Geodatabase 通过定义域 ( Domain ) 和验证规则 ( Validation rule ) 来增强属性的完整性
- Geodatabase 将要素的“自然”行为绑定到存储要素的表中
- Geodatabase 可以具有多个版本 ( Version ),同一时刻允许用户编辑对同一数据进行编辑

### 1.5.2 个人 Geodatabase ( personal Geodatabase ) 和多用户 Geodatabase

Geodatabase 可以分成两种——个人 Geodatabase 和多用户 Geodatabase。

ArcGIS Desktop 对个人 Geodatabase 操作有全功能支持。个人 Geodatabase 适用于小型项目的地理信息系统。Personal Geodatabase 可以使用 Access 数据库打开。在安装 ArcInfo 软件的同时,微软 Jet 引擎也同时完成安装,这样 ArcInfo 便可以创建和更新 Access 数据库。

对于大型的企业数据库,可以使用 ArcSDE 软件。ArcSDE 安装管理企业数据的关系数据库的服务器上。通过 TCP/IP, ArcSDE 为运行在 PC 上的 ArcInfo 应用程序提供 Geodatabase。ArcSDE 可以运行在 Windows NT 或者在 UNIX 操作系统上。

ArcSDE 不仅提供对地理数据的远程访问,而且还允许多用户同时编辑同一地理数据。ArcSDE 可以实行海量数据的中央化管理。

### 15.3 开放且可扩展的数据库服务器

在 ArcInfo 中,你可以根据自己的需要,使用各种大小规格的 Geodatabase。如果数据量不大,你可以直接在 ArcCatalog 中使用个人 Geodatabase。一般来说,个人 Geodatabase 支持的对象少于 250,000 个。它只支持同一时刻的单一编辑用。

如果用户机构数据量巨大,并且有对同一数据进行同时编辑的需求,那么这时候便需要在关系数据库中配置 ArcSDE 了。

使用 ArcSDE 软件的优点:

- 可以有弹性的选择数据库的规模和大小
- 可以使用关系数据库进行空间数据的存储

- 可以在 Windows NT 上或 UNIX 上提供地理数据服务
- 可以向其它的应用程序比如：MapObject，ArcIMS，ArcView GIS 和 CAD 客户端应用程序提供数据服务
- 可以中央化存储和管理 Geodatabase
- 可以集成与 OGC 标准相符的应用程序。
- 可以使用结构化查询语句（SQL）应用程序来访问 Geodatabase 的表和记录

### Geodatabase 提示：

Personal Geodatabase 的功能集成在 ArcInfo 软件中。

ArcSDE 将 geodatabase 数据格式的功能和扩展了空间功能的关系型数据库有机结合。

ArcSDE 可以理解为 ArcInfo 的多用户扩展。

## 1.6 地理数据访问

开发人员可以从三个基本层次中获取 Geodatabase 数据：

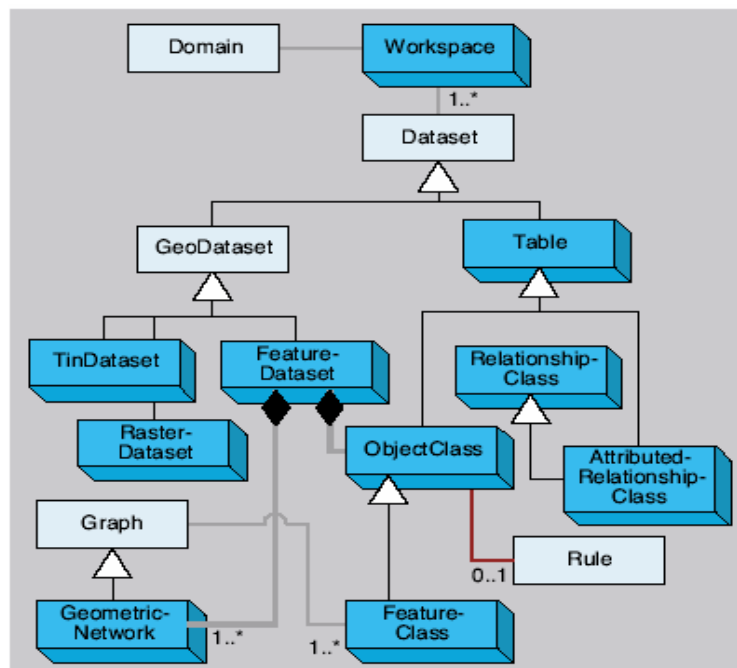
- 通过 Geodatabase 数据访问对象，它是 ArcObject 的一个子集。ArcObject 是 ArcMap 和 ArcCatalog 软件构建的组件基础。
- 简单的非拓扑要素层次，通过符合 OGC 简单要素协议的 ArcSDE 应用程序编程接口实现。
- 行，列和表的层次上，通过关系数据库的本地 SQL 接口实现。

### 1.6.1 通过 ArcObjects 访问数据

获取数据的最合适的方法是通过 Geodatabase 数据访问对象。在这个层次上，Geodatabase 的整个结构被展示出来：拓扑，关系，集成规则和行为，同时还有栅格，表面和位置显示。

你可以通过 ArcObjects 使用 VBA 或 Visual C++ 或其它合适的 COM 开发环境编程获取数据。

以下是一部分 Geodatabase 数据访问对象的一个简化的标准建模语言（UML）表，它将在第四章“地理数据的结构”中被讨论。



## 1.6.2 访问简单要素的数据

对于空间应用程序来说，完全可以获取简单非拓扑要素形式的地理数据。

ArcSDE 下在支持 C API 和 Java API，这两种语言都符合 OGC 简单要素规范。

OGC 是领导空间数据厂商的组织，它的目的是开发标准的软件接口，这样可以使得各种 GIS 空间数据可以自由地进行交换。

在网络上拥有不同格式的地理数据的不同组织可以将本地数据格式组织为“简单要素”集成到应用程序中。

ESRI 是 OGC 技术规范最主要的贡献者，并且承担地理数据公开交换的义务。

## 1.6.3 通过 SQL 访问数据

GIS 是丰富的地理数据资料档案库，这个资料档案库中管理存储了各种自然要素、交通运输之类的基础设施、或者入公共事业几何网络等等这样的数据。使用 Geodatabase 来维护和管理数据的时候，外面的数据库应用程序可以有效地获取享的以非空间形式表达的数据。

使用关系数据库自带的 SQL 接口，便可以建立从 Geodatabase 中提取数据的应用程序，可以将这些提取的数据用于各种分析工作，比如处理工作流程或则统计分析等。

在这个视图中，Geodatabase 是一系列的表、列和行。通过 SQL 接口，可以观察到 Geodatabase 数据库的内部物理结构。这个结构在 ArcInfo 中是不能直接可见的，要通过 ArcCatalog 图形界面管理。使用 SQL 开发程序可以更新要素属性，在这个过程中，尤其小心不要破坏 Geodatabase 的管理结构。

# 1.7 创建数据模型

设计 Geodatabase 与其它数据库原理是基本上一致的。因为 Geodatabase 属于关系数据库的一种，只不过 Geodatabase 中存储的是地理数据。

Geodatabase 通过描述一个基于对象的数据结构，扩展并简化设计流程，在这个结构中，描述了地理要素的空间和拓扑关系。这个结构中有一部分是专门用于将地理对象集成表达为有机系统的，如：河流和道路网络、或者地块集。这样的结构叫做拓扑。

Geodatabase 数据模型构建了“人们对现实世界周围对象认识的理解”和“这些对象在关系数据库中如何存储”的桥梁。

## 1.7.1 Geodatabase 设计

一般来说，关系数据库设计需要两个基本步骤：数据逻辑模型的结构化和数据模型的物理实现。

逻辑数据建模体现数据的用户视图，数据库建模则在关系数据库技术的框架中物理实现数据模型。

## 1.7.2 数据逻辑模型设计

建立数据逻辑模型的关键任务是准确定义有意义的对象集并明确它们之间的关系。

你考虑的对象，可能是比如街道、地块、所有者和建筑物这样的事物。那么对应它们之间关系的一些描述就是“位于”、“被拥有”和“是一部分”等。

在原始逻辑数据模型设计完成后，可以提交给用户，进行下一进程的需求分析，确定用户对数据录入、更新和访问的要求。

设计逻辑数据模型是个反复的过程，通过不断的实践可以获取到最终的“艺术”成果。

不存在绝对“正确”的模型，只能说某个模型适合还是不适合。要确认数据模型是否完全符合需求是非常困难的！可以回答以下三个问题，如果你的答案都是肯定的，那么意味着你所创建的模型已经很不错了：

- 逻辑数据模型表现了所有数据，且没有冗余
- 逻辑数据模型支持业务流程？
- 逻辑数据模型满足不同用户组的数据视图？

### 1.7.3 表达数据逻辑模型

先前，逻辑模型通常绘制为 E-R 图。面向对象技术建模的前人推出了各种不同的设计方法和图表符号。E-R 图的一个不足之处是它们的表现形式会被设计方法左右。

现今，大部分面向对象的建模者已经开始采用统一建模语言（UML），UML 是表达对象模型的标准符号。UML 已被主导软件公司和数据库厂家所认可。

需要提醒的是，UML 不是设计方法，而是一个绘图符号。UML 支持面向对象的设计方法，并且按照标准方式表达数据模型。

本书是使用 UML 绘制 ArcInfo 对象模型图（也称为 ArcObjects）的。同样，如果需要绘制定制的 Geodatabase 对象模型，也使用 UML。

### 1.7.4 物理模型实现

物理库模型由逻辑模型创建而得。一般情况下，关系型数据库专业开发者从数据建模者那儿得到逻辑模型后，使用数据库管理工具来定义数据库方案（schema）并创建新数据，为数据输入做好准备。

数据库物理设计跟逻辑模型设计虽然有很多相似之处，但是也存在很多区别。逻辑模型中设计好的类和对象在物理表中实现的时候经常会被拆分或者合并。规则和关联类也可以使用多种方式表达。

Geodatabase 的一个显著优势在于它不是仅仅实现数据的物理配置而已，你可以在 Geodatabase 中构建与逻辑模型相近的物理结构。

### 1.7.5 逻辑模型和数据库模型中的元素

下图中表示的是数据逻辑模型中的元素以及它们相应的数据库中的元素：

Logical elements	Database elements
Object	Row
Attribute	Column, Field
Class	Table

逻辑模型是对人类生产应用中所处理的对象的一种抽象。这种抽象被具体成数据库中的元素。

比如，对象（Object）都代表某个实体，如一栋房子、一条河流，或消一名费者。对象用来组织存储。对象具有一系列的属性。属性值体现对象的特征，比如对象名称、度量值、等级。属性以列（或字段）的形式存储在一个数据库中。

类 (Class) 是相似对象的集合, 类中的每一个对象都具有相同的一组属性。类以表的形式存储在数据库中。表中的行、列构成一个二维矩阵。

### 1.7.6 复杂数据处理

由于关系数据库是基于简单、智能和易于理解的理论基础建立的, 所以在商业推广上, 它们具有优势。当然, 这种简化可以理解为优点, 也可以看作是短处——建立关系数据库的概念是很简单的, 但是在对复杂数据建模的时候, 就会有些困难。

Geodatabase 包含有复杂的地理数据。构成线要素和面要素的一系列坐标形状是不能用标准类型的字段 (如: 整形, 实数, 字符串) 来表示的。并且, 在地理数据库中, 要素一般都是具有一定的拓扑关联、隐含的空间关联或者普通关联的。

关系数据库是 Geodatabase 的基础。Geodatabase 的目的是使用独立于关系数据库的统一的数据模型来处理复杂的地理数据。本书第十二章, “Geodatabase 设计指南”, 将进一步介绍 Geodatabase 设计方法。

### 1.7.7 Geodatabase 设计原则

使用 Geodatabase, 你可以设计和地理数据库的逻辑概念模型非常相近的数据库——Geodatabase 是用要素集、要素类、拓扑关系、关联类和其它元素组织的。对于 ESRI 的数据模型的开发人员来说, 这也是在 ArcInfo 中引入 Geodatabase 数据模型的基本原因。

以下是 Geodatabase 设计的基本步骤:

1. 数据的用户视图建模。和用户交流, 了解用户结构, 解析用户的业务模型。
2. 定义对象和关联。使用对象集和创建逻辑数据模型, 并定义它们之间的关联。
3. 选择地理描述方式。对有意义的数据采用确切的描述方式, 如矢量、栅格、表面或 Locator。
4. 与 Geodatabase 元素匹配。将逻辑模型中的对象与 Geodatabase 中的元素互相匹配起来。
5. 组织 Geodatabase 结构。考虑地图专题、拓扑关联以及不同用户部门对数据的需求, 创建 Geodatabase。

编译: 张俊、孙朝阳、庄怀耀

策划、审校: 邢超

主编: 曾杉