

编者按: Oracle Spatial 是 MapInfo 公司和 Oracle 公司合作开发的一个能在 Oracle 数据库中快速有效地存储、访问、分析空间数据的完整的功能程序集。该程序可为农业信息系统的构建提供参考, 从而设计出集信息采集、发布、处理、存储、开发应用为一体的综合应用系统。

利用 MapX 访问 Oracle Spatial 空间数据库的研究

崔伦辉, 张万昌

(1. 南京大学国际地球系统研究所, 江苏南京210093; 2. 中国科学院大气物理研究所东亚区域气候—环境重点实验室, 北京100029)

摘要 在详细介绍 Oracle Spatial 空间数据存储模型的基础上, 阐述了利用 EasyLoader 工具上传空间数据的原理以及利用 MapX 访问 Oracle Spatial 空间数据库的方法, 并针对利用 EasyLoader 工具上传空间数据时造成的部分信息丢失问题提出了有效的解决方案。

关键词 MapX; Oracle Spatial; EasyLoader; 数据上传

中图分类号 S126 文献标识码 A 文章编号 0517-6611(2007)31-10158-03

Accessing Oracle Spatial Database Using MapX

CUI Lunhui et al (International Institute for Earth System Science, Nanjing University, Nanjing, Jiangsu 210093)

Abstract On the basis of systematic introduction on the storage model of spatial database in Oracle Spatial, principles of spatial data uploading by means of EasyLoader and the methodology for accessing spatial database in Oracle Spatial by using MapX technology were proposed and discussed, an effective solution for preventing information lost in the process of spatial data uploadings with Easyloader under the proposed scheme was given.

Key words MapX; Oracle Spatial; EasyLoader; Data Upload

Oracle Spatial 是 MapInfo 公司和 Oracle 公司合作开发的一个能在 Oracle 数据库中快速有效地存储、访问、分析空间数据的完整的功能程序集。它利用 SDO_GEOMETRY 字段存储空间数据, 实现了空间数据和属性数据的一体化存储; 利用 USER_SDO_GEOM_METADATA_TABLE 视图管理空间数据表; 采用 R-Tree 索引和四叉树索引提高访问速度, 同时提供了大量的空间分析函数以实现空间分析功能。

MapX 组件是 MapInfo 公司开发的具有强大地图分析功能的 ActiveX 组件, 利用 MapX 能够将地图嵌入到应用程序中。它的主要功能包括地图的放大、缩小、漫游、选择等, 图层的自由控制, 动态图层创建和图层自定义, 专题图制作, 地理查询等。

为了便于上传空间数据, MapInfo 公司提供了 EasyLoader 工具。它能够方便地将 tab 数据上载到数据库中, 然而由于 MapInfo 和 Oracle Spatial 所支持的数据类型不一致, 在上载过程中有部分图元信息丢失, 如圆角矩形、弧、椭圆等图元的坐标、样式信息和矩形的坐标信息不能上载。该文利用点的坐标信息来代替矩形、弧、椭圆等图元的坐标信息, 然后再重新生成新的矩形、弧、椭圆等图元的方法解决了这一问题。具体的转换方法需从 Oracle Spatial 中空间数据的存储机制谈起。

1 Oracle Spatial 存储机制

Oracle Spatial 利用 SDO_GEOMETRY 字段存储空间数据, SDO_GEOMETRY 是 OpenGIS 规范定义的一个对象, 它由 5 个字段组成, 分别为: SDO_GTYPE、SDO_SRID、SDO_POINT、SDO_ELEM_INFO、SDO_ORDINATES。

1.1 SDO_GTYPE SDO_GTYPE 是一个 NUMBER 型的数

值, 用来定义存储对象的类型。SDO_GTYPE 是由 4 个数字组成的整数, 其格式为 dtt, 其中“d”表示几何对象的维数; “1”表示三维线性参考系统中的线性参考值, 当“d”为三维或四维时需要设置该值, 一般情况下为空; “tt”为几何对象的类型, Oracle Spatial 定义了 7 种类型的几何对象(表 1)。目前, “tt”使用了 00~07 (包括一种用户自定义类型), 而 08~99 是 Oracle Spatial 保留的数字, 以备将来几何对象扩展时使用。

表 1 Oracle Spatial 支撑的几何对象类型

数值	几何类型	说明
d100	未知类型	用于存放自定义类型几何对象, 如样条曲线
d101	点	几何对象包含 1 个点
d102	直线或弧线	几何对象由直线段和弧线段组成
d103	多边形	几何对象包含一个多边形, 该多边形可以含有洞
d104	多种形状集合	点、线、多边形超集, 可包含所有类型
d105	复合点	几何对象包含 1 个或多个点, 是点类的超集
d106	复合线	几何对象包含 1 个或多个线串
d107	复合多边形	几何对象可以包含多个外环、多个不相交的多边形

1.2 SDO_SRID SDO_SRID 也是一个 NUMBER 型的数值, 它用于标识与几何对象相关的空间坐标参考系。

1.3 SDO_POINT SDO_POINT 是一个包含 X、Y、Z 数值信息的对象, 用于表示几何类型为点的几何对象。

1.4 SDO_ELEM_INFO SDO_ELEM_INFO 是一个可变长度的数组, 每 3 个数作为一个元素单位, 用于解释坐标如何存储在 SDO_ORDINATES 数组中。它包括 SDO_STARTING_OFFSET、SDO_ETYPE 和 SDO_INTERPRETATION 3 个字段。其中, SDO_STARTING_OFFSET 表示每个几何元素的第 1 个坐标在 SDO_ORDINATES 数组中的存储位置。SDO_E

TYPE 和 SDO.INTERPRETATION 联用表示图元,其代表的意义很多(表2)。

1.5 SDO.ORDINATES SDO.ORDINATES 是一个可变长度的数组,用于存储几何对象的真实坐标。该数组的类型为 NUMBER 型,它的最大长度为 1 048 576。SDO.ORDINATES 必须与 SDO.ELEM.INFO 数组配合使用,才具有实际意义。

表2 DO.ETYPE 和 SDO.INTERPRETATION 的关系

SDO.ETYPE	SDO. INTERPRETATION	含义
0	任意值	用于为 Oracle Spatial 不支持的几何类型建模。
1	1	点
1	n>1	具有 n 个点的点集。
2	1	由直线段组成的线串,一个直线段由起点和终点组成。
2	2	由弧线组成的线串,一个弧线段由起点、弧上任意一点以及终点组成。相邻两弧段的接点无需重复存储。
1003 或 2003	1	由直线段组成的多边形,起点与终点必须相同。
1003 或 2003	2	由弧线段组成的多边形,起点与终点必须相同。一个弧线段由起点、弧上任意一点以及终点组成。相邻两弧段的接点无需重复存储。
1003 或 2003	3	矩形,由左下角和右上角两点确定。
1003 或 2003	4	圆,由圆周上的不同三点确定。
4	n>1	由直线段和弧线段组成的复合线, n 表示组成复合线的相邻子元素的个数,子元素的 SDO.ETYPE 必须为 2,一个子元素的最后一点是下一个子元素的第一个点,并且该点不能重复。
1005 或 2005	n>1	由直线段和弧线段组成的复合多边形, n 表示组成复合多边形的相邻子元素的个数,子元素的 SDO.ETYPE 必须为 2,一个子元素的最后一点是下一个子元素的第一个点,并且该点不能重复。多边形的起点和终点必须相同。

2 利用 EasyLoader 工具上载空间数据

利用 EasyLoader 工具上载空间数据方便快捷,系统做了很多重要的工作,主要包括: 如果是第 1 次上载 tab 表,在数据库中将自动创建 MapInfo 用户,并在该用户下创建 MAPINFO.MAPCATALOG 表和 MAPCATALOG.IDX 索引,该表用来记录上传到数据库中的地图信息,包括表名、属主名、边界、坐标、样式等信息; 数据表上传之后将在表中自动添加 M.STYLE、M.PRINX、GEOLOC 3 个字段。M.STYLE 用来记录图元的样式信息;M.PRINX 为图元的唯一索引;GEOLOC 存储图元坐标数据。同时系统将在 M.PRINX 和 GEOLOC 上分别创建索引; 将表名、空间列、采用的坐标系统等信息添加到 MDSYS 用户下的 SDO.GEOM.METADATA.TABLE 表中。

3 不支持数据类型的上传方法

由于 MapInfo 和 Oracle 所定义的数据类型稍有不同,在数据转换时将有一部分信息不能被转换,表3 列出了空间类型的转换关系。

从表3 中可以看出,椭圆、弧线、矩形等尽管在 Oracle 中有定义,但是不能上传。为解决这一问题,该文在原图层中手工添加一些点来提取坐标信息,利用程序再转化为原来的图形后,再删除这些手工添加的点。上传之前需在表中添加 3 个字段:flag(用于记录是否为手工添加的点,如果是则为 1) lname(用于记录该点是哪一图元的离散化点,同一图元的离散化点 lname 相同) type(标记类型,如矩形为 1、圆

角矩形为 2、弧线为 3 等)。这样每个图元的信息都可由这些离散化点转化得到(图1)。在 Non-Earth(kilometers) 坐标系下具体方法如下:

表3 MapInfo 和 Spatial 数据类型

MapInfo 类型	Oracle 类型	含义
Point	POINT	点
Line	LINESRING	线
Region	POLYGON	多边形
Polyline	MULTILINESTRING	多线段的折线
Region with multiple polygons	MULTIPOLYGON	多个多边形的集合
Ellipse	NULL	椭圆
Arc	NULL	弧
Rectangle	NULL	矩形
Text	NULL	文本
Round Rectangle	NULL	圆角矩形

3.1 矩形 取左下角和右上角的 2 个点,设置 flag 为 1, lname 为 REC,type 为 1(代表矩形)。程序在运行时首先选择名为 REC 的点,然后根据 type 的值确定 2 个点代表矩形,再修改表中的 GEOLOC 字段,将其值修改为 SDO.GEOMETRY(2003,262147,NULL,SDO.ELEM.INFO.ARRAY(1,1003,3),SDO.ORDINATE.ARRAY(x₁,y₁,x₂,y₂))。x₁ 为左下角点的 x 坐标,y₁ 为左下角点的 y 坐标,x₂ 为右上角点的 x 坐标,y₂ 为右上角点的 y 坐标。

3.2 圆角矩形 取圆角矩形 4 条边上的 4 个点,根据这 4 个点的坐标计算出圆角和线段上 12 个点的坐标,用弧线和线段连接围成的闭合多边形来表示圆角矩形。GEOLOC 字段的值修改为 SDO.GEOMETRY(2003,262147,NULL,SDO.ELEM.INFO.ARRAY(1,1005,8,1,2,2,5,2,1,9,2,2,13,2,1,17,2,2,21,2,1,25,2,2,29,2,1),SDO.ORDINATE.ARRAY(...))。SDO.ORDINATE.ARRAY 中省略号代表按顺序存储的各点的实际坐标值,由于点较多故省略,以下 SDO.ORDINATE.ARRAY 中点坐标值的标号都是按顺时针顺序排列的。

3.3 圆 取圆上 3 点,根据这 3 点的坐标生成圆。GEOLOC 字段的值修改为 SDO.GEOMETRY(2003,262147,NULL,SDO.ELEM.INFO.ARRAY(1,1003,4),SDO.ORDINATE.ARRAY(x₁,y₁,x₂,y₂,x₃,y₃))。

3.4 椭圆 取椭圆长轴短轴上的 4 点,用 2 条弧线围成的多边形来表示椭圆。GEOLOC 字段的值修改为 SDO.GEOMETRY(2003,262147,NULL,SDO.ELEM.INFO.ARRAY(1,1003,2,1,2,2,5,2,2),SDO.ORDINATE.ARRAY(x₁,y₁,x₂,y₂,x₃,y₃,x₄,y₄))。

3.5 弧 取极点和拐点以及两者之间的点,将它们连接成弧。GEOLOC 字段的值修改为 SDO.GEOMETRY(2002,262147,NULL,SDO.ELEM.INFO.ARRAY(1,2,2),SDO.ORDINATE.ARRAY(x₁,y₁,x₂,y₂,x₃,y₃))。

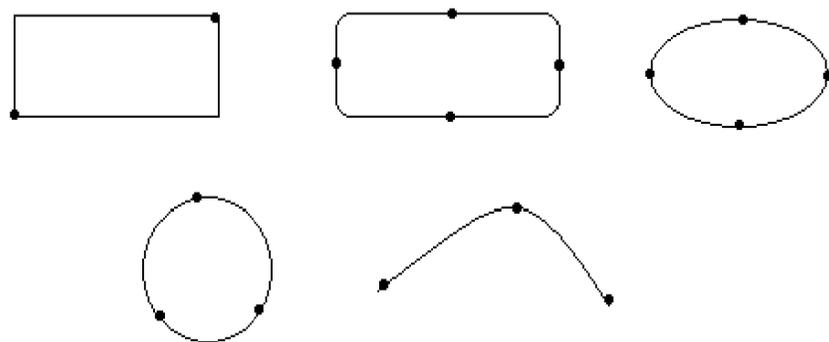


图1 各图元的离散化点

最后,删除所有flag 值为1 的点,这样,flag、type、lname 等字段的值都为空,修改表结构删除这3 列,上载到数据库中的表的表结构中不存在多余字段。矩形、圆角矩形、圆、椭圆仍为面结构,圆弧仍为线结构,保持了图元的结构信息,为以后空间分析函数的应用提供了可能。

4 MapX 访问 Oracle Spatial 方法

MapX 有两种方式访问 Oracle 空间数据,即 ODBC 方式和 OCI 方式。ODBC 不仅需要配置数据源 DNS,同时效率不高;OCI 方式是底层接口,速度快,该文采用 OCI 方式。通过 OCI 访问空间数据相当于从服务器端下载相关的空间数据并在客户端进行显示,客户端显示空间数据的方式就是在原有地图上增加一层新的地图。因此,首先要创建一个新的图层,然后设置连接参数从服务器下载数据,再添加图层进行显示。C# 语言程序如下:

```
MapXLib.LayerInfo lyr ;
lyr = new MapXLib.LayerInfoClass() ;
lyr.Type = MapXLib.LayerInfoConstants.miLayerInfo-
TypeServer ;
lyr.AddParameter( "Name", "house" ) ;
lyr.AddParameter( "ConnectionString", "SRVR = server name ;
UD = user name ;PWD = password" ) ;
```

```
lyr.AddParameter( "Query", "select * fromtable name" ) ;
lyr.AddParameter( "Tool kit", "ORAINET" ) ;
axMap1.Layers.Add(lyr, 1) ;
```

5 小结

基于 Oracle Spatial 空间数据库的 MapX 应用程序的开发,集成了 MapX 和 Oracle Spatial 的优势。然而,由于两者所支持的数据类型不一致,在数据转换时会有部分信息不能转换,造成数据丢失。为解决这一问题,该文避开了读取 MapInfo 数据结构的繁琐,在上载之前对个别图元用点进行离散化,上载后运行一段程序代码就可实现数据的全部上载,方便操作,提高了效率。

参考文献

- [1] Oracle Corporation. Oracle Spatial Users Guide and Reference[M]. 2002.
- [2] 霍宏,胡福乔.用 MapX 访问 Oracle 中的空间数据[J]. 计算机应用, 2003, 23(1) :113- 114.
- [3] 齐锐, 屈韶琳, 阳琳.用 MapX 开发地理信息系统[M]. 北京: 清华大学出版社,2003.
- [4] 赵志民, 曹巨辉, 郑雷. 基于 Oracle Spatial 技术开发 MapX 应用程序[J]. 海洋测绘,2004, 24(4) :36- 38.
- [5] 芮小平, 张彦敏. Oracle Spatial 几何类型字段解析[J]. 物探化探计算技术,2004, 26(4) :359- 362.
- [6] 郑军, 陈正阳. 利用 OCI 与 OLE 开发 Oracle 空间数据库[J]. 计算机工程与设计,2004, 25(6) :1007- 1009.