**引文格式**:吴政,李成名,武鹏达,等.Oracle 数据库矢栅数据一体化存储与管理[J].测绘学报,2017,46(5):639-648. DOI:10.11947/j.AGCS.2017.20160586.

WU Zheng, LI Chengming, WU Pengda, et al. Integrated Storage and Management of Vector and Raster Data Based on Oracle Database[J]. Acta Geodaetica et Cartographica Sinica, 2017, 46(5): 639-648. DOI: 10.11947/j.AGCS.2017.20160586.

# Oracle 数据库矢栅数据一体化存储与管理

吴 政,李成名,武鹏达,沈建明,孙 伟

中国测绘科学研究院 GIS 所,北京 100830

# Integrated Storage and Management of Vector and Raster Data Based on Oracle Database

WU Zheng, LI Chengming, WU Pengda, SHEN Jianming, SUN Wei

Institute of GIS, Chinese Academy of Surveying and Mapping, Beijing 100830, China

**Abstract**: At present, there are many problems in the storage and management of multi-source heterogeneous spatial data, such as the difficulty of transferring, the lack of unified storage and the low efficiency. By combining relational database and spatial data engine technology, an approach for integrated storage and management of vector and raster data is proposed on the basis of Oracle in this paper. This approach establishes an integrated storage model on vector and raster data and optimizes the retrieval mechanism at first, then designs a framework for the seamless data transfer, finally realizes the unified storage and efficient management of multi-source heterogeneous data. By comparing experimental results with the international leading similar software ArcSDE, it is proved that the proposed approach has higher data transfer performance and better query retrieval efficiency.

**Key words**: multi-source heterogeneous spatial data; vector and raster data integrated storage; retrieval optimization; data transfer

**Foundation support**: The National Key Technology Research and Development Program of the Ministry of Science and Technology of China (No.2015BAJ06B01); National Basic Surveying and Mapping Project (Nos. A1608;A1615)

**摘 要**:当前多源异构空间数据存储和管理存在流转难度大、存储不统一、效率不高等问题。本文将关系数据库与空间数据引擎技术结合,提出了一种基于 Oracle 的面向对象的矢栅数据一体化存储与管理方法。该方法首先建立了矢栅一体化存储模型并优化了检索方法,然后据此构建了数据无缝流转框架,实现了多源异构矢栅数据的统一存储与高效管理。通过与国际领先的同类软件 ArcSDE 进行测试对比,该方法具有更高的数据流转性能及更优的查询检索效率。

**关键词**:多源异构空间数据;矢栅一体化存储;检索优化;数据流转

**中图分类号**:P208 **文献标识码**:A **文章编号**:1001-1595(2017)05-0639-10

**基金项目**:国家科技支撑计划(2015BAJ06B01);国家基础测绘项目(A1608;A1615)

随着空间地理信息技术的不断发展,获取地理空间信息的手段不断丰富,地理空间数据呈现出数据来源多样化、数据模型差异化、数据存储分散化、空间关系复杂化、数量海量化等特点[1-2],在此情形下,如何高效地存储、组织、管理多源异构的空间数据逐渐成为地理信息系统考虑的核心问题。

以数据、应用为导向的各种空间数据库技术、空间数据引擎技术、分布式存储管理技术在一定程度上或从某一方面有效地解决了地理空间数据存储和管理的问题。目前,国内广泛使用的数据库主要集中在以 Oracle 为代表的“扩展对象关系型数据库”、“空间数据引擎”两方面。然而,多年的数据库建设涌现了种类繁多、架构各异的 GIS

系统,多源异构的空间数据出现难以共享、迁移效率低、流转难度大、互操作难以实现等问题。基于此,本文针对矢量与栅格数据这两种主要的空间数据结构,提出一种基于 Oracle 的面向对象的矢栅一体化管理方法,建立了矢栅数据一体化存储模型,优化了矢栅数据的检索方法,并据此设计了多源异构数据流转框架,达到对空间数据进行统一存储、统一访问、统一管理的目的。

## 1 矢栅数据存储和管理的现状及问题

针对多源异构矢栅数据的存储、组织、管理与发布等问题,国内外已经有了大量的研究成果,主要分为三大类[3-6]:第一类是以传统关系型数据库为代表,直接对传统数据库管理系统进行扩展,实现矢栅数据存储与管理功能;第二类是基于传统数据库管理系统开发空间数据引擎,以获得传统数据库管理系统功能之外的矢栅数据存储和管理能力;第三类是基于分布式文件系统构建非关系型(NoSQL)数据库,在分布式并行计算框架下进行数据处理。

### 1.1 基于扩展对象的关系型数据库

该类关系型数据库通过对自身的数据库管理系统进行扩展,使之能直接存储和管理非结构化的矢栅数据,如 Oracle 公司的 Oracle Spatial、Oracle GeoRaster、Postgres SQL 的 PostGIS、IBM 公司的 DB2 Spatial Extender、Informix 的 Spatial DataBlade 等。这些传统的数据库产品其技术成熟稳定,在各行业应用范围广、用户数量多、运行的周期长,基于其部署的应用系统数量和规模巨大,其中以 Oracle 应用最为广泛。

该类数据存储的共同问题是需要将空间几何信息转化为各自扩展模块内部专有的存储对象,数据存储模型的差异导致无法直接与其他数据源或数据库进行数据共享或互操作,需要通过中间件进行矢栅数据格式的转换,不具有普适性。如 Oracle 为矢量数据提供了 Oracle Spatial 扩展模块[7],为栅格数据提供了 GeoRaster 扩展模块[8],都需要将空间几何信息转换为其对应的内部存储对象,不利于共享和互操作;同时,GeoRaster 在栅格数据管理方面也存在限制:如栅格数据导出格式的限制,单次栅格数据导出大小的限制,栅格数据存取时需要借助第三方工具等。

### 1.2 基于传统关系型数据库的空间数据引擎

空间数据引擎是基于传统关系型数据库管理系统开发的支持空间数据的中间件,采用的是客户/服务器体系结构,应用程序通过空间数据引擎构建矢栅数据模型,将数据交给数据库统一管理,具有代表性的是 ESRI 的 ArcSDE、MapInfo 的 Spatial-Ware、MAP GIS 的 MapGIS SDE、超图的 SuperMap SDX 等,其中基于 ArcSDE 应用开发的案例最多,处于国际领先地位。

该类数据存储的主要问题是支持的数据源有限,在大批量的数据入库及数据迁移时效率较低,同时其部署时需要在数据库服务器端安装相应的 SDE 环境,对未通过 SDE 入库的数据无法直接读取,无形中增加了数据库服务器端负荷,无法实现多源异构数据的共享与数据的流转。

### 1.3 基于分布式文件系统的非关系型数据库

目前国内外针对海量空间数据存储和管理的研究主要集中在基于 HBase、MongoDB 等 NoSQL 数据库构建结构化(Key-Value)存储[9-10],实现对高分辨率遥感影像、海量矢量数据的管理。如文献[11]提出了一种海量空间数据处理并行框架(IDEA),可以有效地分配数据服务和计算节点,实现任务的有效调度;文献[12]提出了基于整合的 Hadoop 和 Eucalyptus 云平台的高分辨率遥感影像存储模型 C-RSM;文献[13]基于 MongoDB 和 Hadoop 架构,设计并实现了矢量空间数据云存储与处理系统 VectorDB,达到了海量矢量数据的高效存储与处理要求。

尽管很多学者基于非关系型数据库提出了空间数据管理的方案,但在商业化应用中使用的并不多,原因包括:针对矢量数据的分布式存储、空间操作的分布式处理等技术并不完善;对多源异构数据还无法提供充足的支持;对于如何集成已存在于关系型数据库中的空间数据,还未有很好的解决方案。

## 2 基于 Oracle 的矢栅数据一体化存储与管理体系架构

为了将矢栅数据及其属性数据、元数据等进行统一存储和管理,便于数据迁移、流转,利于共享,本文根据“异构同化,同构整体化”的思想[14-16]以及多空间数据库系统、分布式数据库系统的模式结构[17-19],遵循数据远程存取、数据操作本地化、兼容多源异构数据的原则[20-21],采用 C/S 体系架构,提出基于 Oracle 的矢栅数据一体化存储与管理方法,用以实现分布式系统多源异

构信息的集成,其体系架构如图 1 所示。

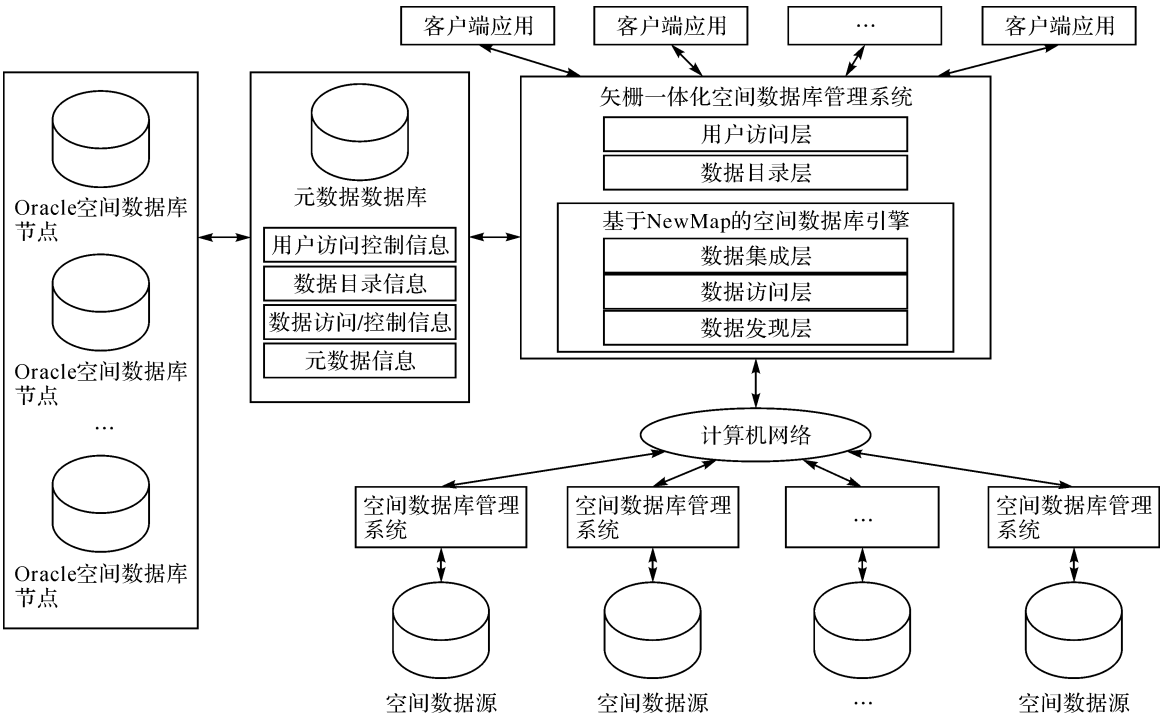



图 1　矢栅数据管理总体架构

Fig.1　Overall framework of vector and raster data management system

该架构采用全局统一的空间数据模型来描述全局概念模式和分片模式[14],实现局部概念模式到全局分片模式的转换,并提供统一的访问接口、统一的空间查询语言和统一的用户交互界面,将分布式的异构矢栅数据存储在各 Oracle 节点中,实现了矢栅数据在各数据库节点之间的无缝流转。参照文献[14]中提出的分布式多空间数据库的集成模型,本文设计了矢栅数据存储和管理的 5 层层次模型(图 1)。

空间数据引擎包含数据发现、数据访问和数据集成 3 层。数据发现层主要解决矢栅数据的获取问题,即通过管理矢栅数据的存储位置、格式、获取及使用方法等元数据信息,为数据引擎提供获取数据的途径;数据访问层主要解决局部异构矢栅数据的转换问题,即实现矢栅数据“异构同化”,将异构的局部矢栅数据模型转换为全局统一的矢栅数据模型,存储在相应的 Oracle 节点中,是数据引擎实现的关键;数据集成层主要解决“物理分布、逻辑统一”的问题,即在不同 Oracle 节点存储的数据在逻辑操作上表现为一个整体,达到“同构整体化”,实现多图层、多数据源的叠加和聚合。其余 2 层为数据目录层和用户访问层。数据目录层主要用于实现数据资源的集中管理;用户访问层主要解决用户统一、透明访问。元数据数据库主要存储了用户访问控制信息、数据目录信息、数据访问控制信息等,为上述 5 层层次模型提供相关元数据信息。

## 3　矢栅数据一体化存储与管理实现方法

### 3.1　矢量数据存储与索引方法

本文对矢量数据的空间几何信息的存储支持两种方式:一种是采用 Oracle Spatial 扩展模块,将空间几何信息转化为 SDO_GEOMETRY 对象进行存储;另一种是以 WKB 的形式存储在 BLOB 对象中。

采用 Oracle Spatial 扩展模块管理矢量空间数据需要对 SDO_GEOM_METADATA(空间元数据表)、空间数据表进行维护和操作。针对矢量空间数据以 BLOB 对象存取方式,本文设计了元数据表、矢量空间数据表、矢量空间索引表、空间参考表 4 类关系表,如图 2 所示。其中,元数据表主要描述空间几何类型、空间坐标范围、空间坐标参考等基本信息,用于数据库内各空间数据基本信息的查询;矢量空间数据表是空间数据的实际

载体,包含空间数据唯一标识、各属性字段、几何信息字段;矢量空间索引表主要描述每个要素的空间范围信息,该逻辑关系表可以用作索引亦可用于创建四叉树、R-Tree 等空间索引或自定义索引;矢量空间参考表主要描述各种空间坐标系的定义及描述,主要用于坐标系统的转换以及空间数据坐标的校准和查询,其包括坐标系统类型、坐标系统的定义。

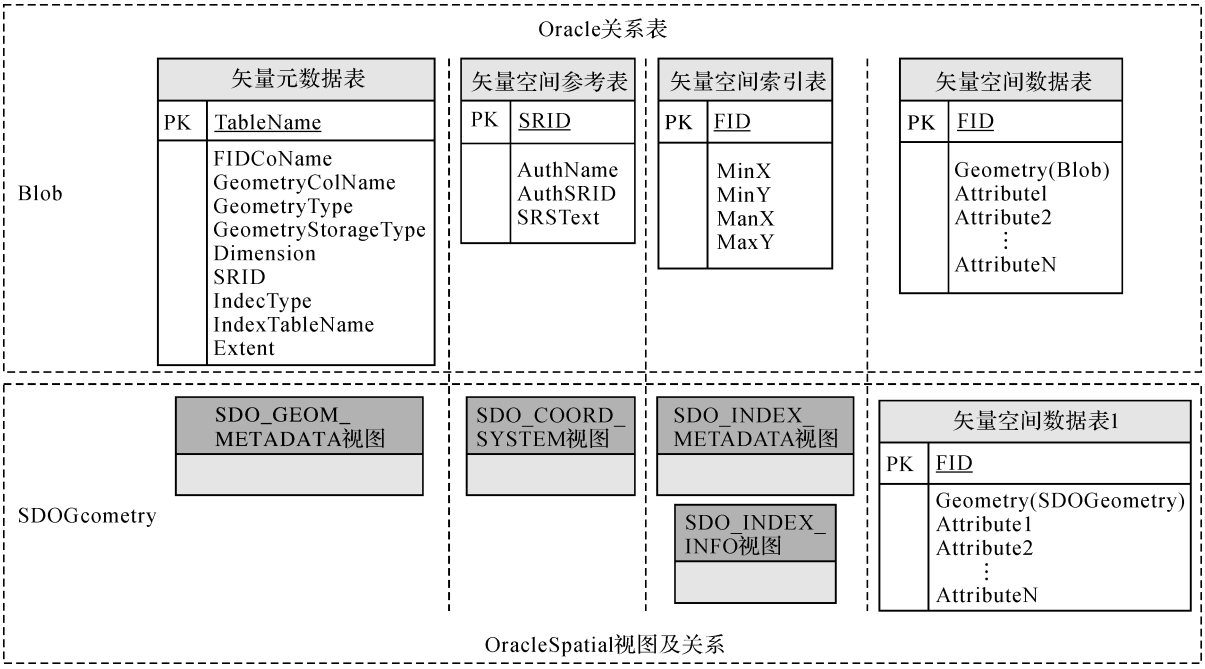



图 2 矢量数据存储关系

Fig.2 Vector data storage relation

为了提高索引机制的可扩展性和灵活性,同时兼顾以 SDO_GEOMETRY 和 BLOB 存储的情况,本文设计了基于 Oracle 的二级索引机制,可以有效兼容不同存储方式的矢量数据检索,如图 3 所示:①对于采用以 SDO_GEOMETRY 存储的空间数据,建立空间索引后,可以直接对 SDO_GEOMETRY 对象进行精确空间查询;②对于以 BLOB 存储的空间数据,首先对矢量空间索引表或索引文件进行空间查询,初步筛选出要素标识 FID(feature identification)结果集,进而依据空间过滤条件对提取的要素进行空间几何运算,得出满足空间过滤条件的对应要素。

### 3.2 栅格数据存储与索引方法

本文采用先按波段分层、再层内分块、最后构建金字塔的思想,设计了具有支持多种栅格数据格式、易于远程存取、提取检索快捷的存储结构。该结构包括:提取栅格数据元数据信息,生成栅格元数据信息表;按照波段对栅格数据进行抽取,并形成波段信息表;对每一波段进行分块,每一块作为一个 BLOB 字段数据存储在栅格数据表中;根据实际需求选择适当的重采样算法,生成影像金字塔,形成金字塔信息表;对生成的金字塔数据采取上述类似方法,生成金字塔数据表,如图 4 所示。

栅格数据存储涉及的关系表如图 5 所示,主要包括栅格信息表、波段信息表、金字塔信息表、栅格数据表、金字塔数据表,各表通过唯一标识(RASTER_ID)进行关联。栅格信息表主要描述了栅格数据波段、像素、空间信息等元数据信息,便于对栅格数据的查询、统计、存取以及格式转换;金字塔信息表主要描述金字塔所属波段号、金字塔级数以及构建金字塔进行块采样的大小等基本信息,主要用于金字塔信息的提取和查询;栅格数据表主要描述栅格数据所属的波段、所在波段中块的位置、块的空间范围信息以及数据块实体;金字塔数据表主要描述金字塔数据所属波段、所属级别、在该级别所属块位置、块的空间范围以及数据实体。本文提出的栅格数据存储结构较完备地存储了栅格数据的基本信息,可以方便地抽取波段信息、快速查找定位空间范围内的栅格数据。

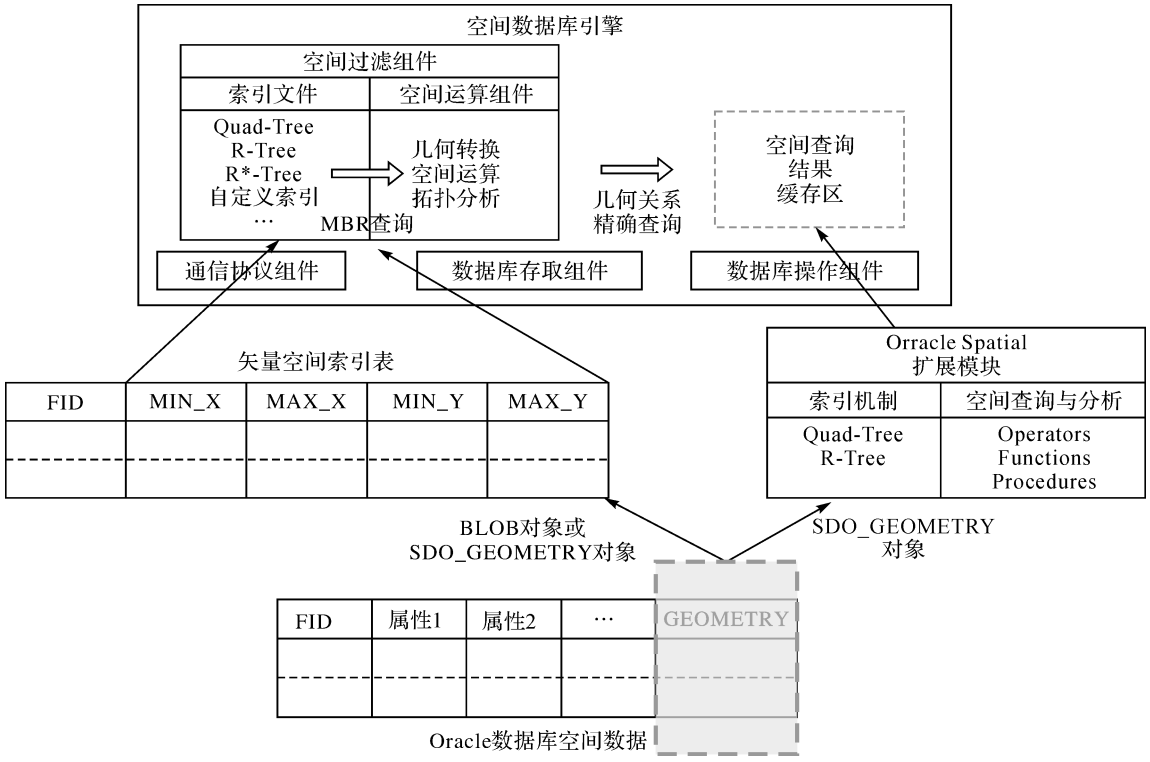



图 3　矢量空间数据查询

Fig.3　Vector spatial data query

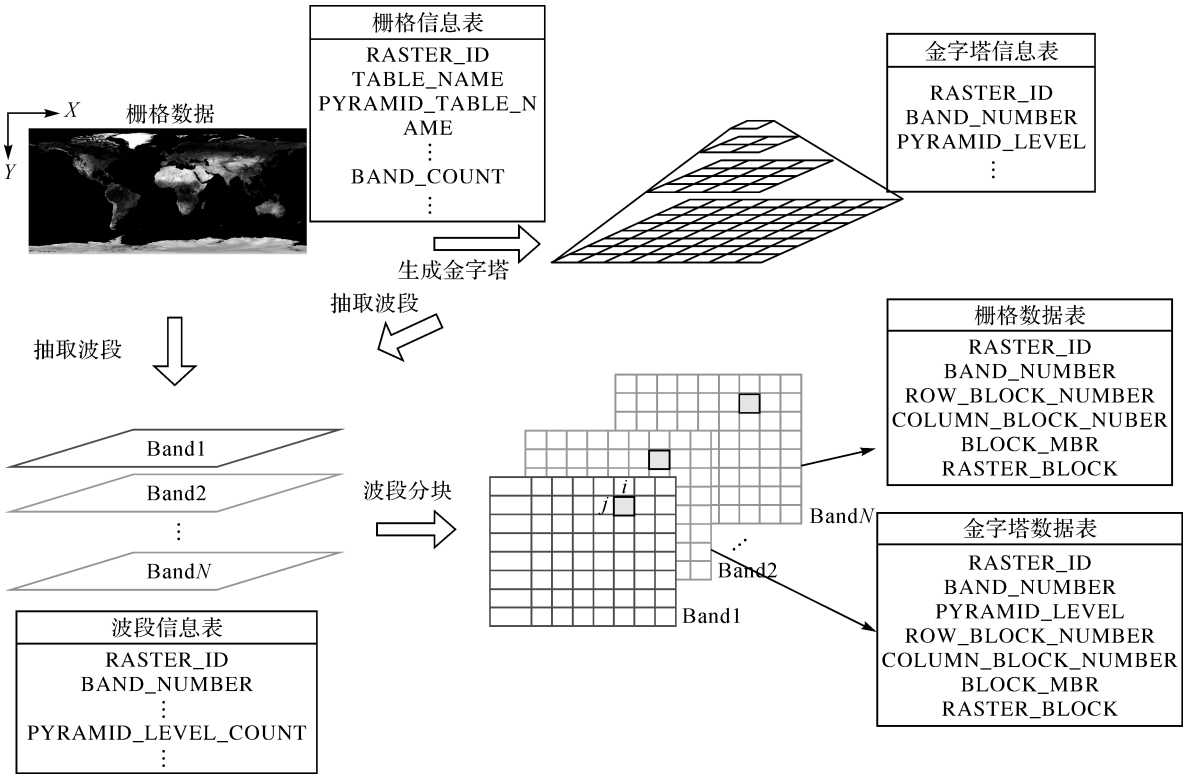



图 4　栅格数据存储结构

Fig.4　Raster data storage structure

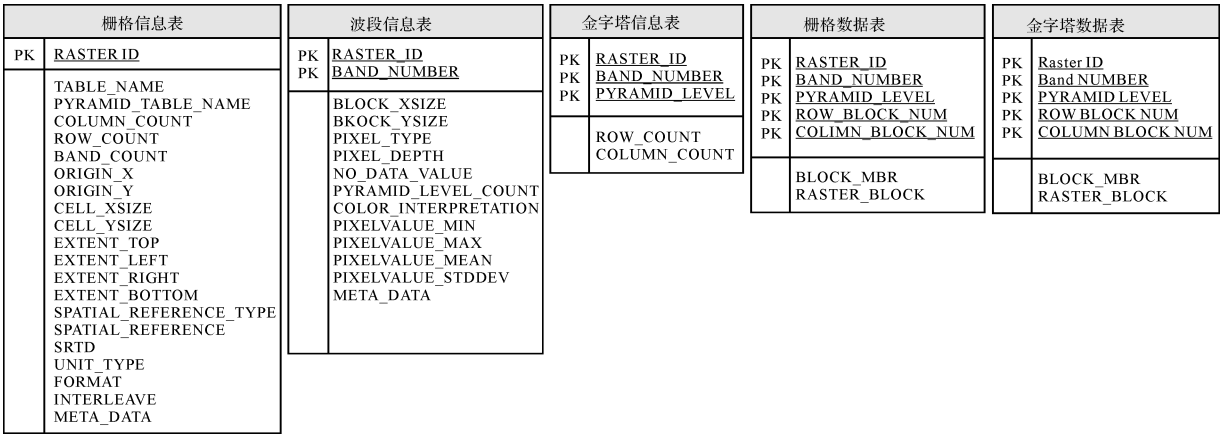



图 5 栅格数据存储关系表

Fig.5 Raster data storage relation table

基于上述提出的栅格数据存储模型，本文建立如下的栅格数据检索方法：

(1) 首先根据栅格数据大小及栅格的地理范围，计算栅格数据的单个像素在平面坐标系中代表的实际地理范围 $cell_x$、$cell_y$；

(2) 获取用户请求的地理坐标范围，结合步骤(1)中的 $cell_x$、$cell_y$，计算当前请求栅格数据的长度像素数 $req_x$ 和宽度像素数 $req_y$；

(3) 根据当前视窗的宽度和高度(屏幕像素数)，结合栅格数据 $req_x$、$req_y$，计算当前视窗范围内的比例尺，找到与比例尺最近一级金字塔级别；

(4) 根据金字塔级别，查找该级金字塔在某一波段内块的位置，见式(1)和式(2)

$$\left.\begin{aligned} tile_{min x} &= \left|\frac{req_{min x}-img_{min x}}{cell_x \times block_x}\right| \\ tile_{min y} &= \left|\frac{req_{min y}-img_{min y}}{cell_y \times block_y}\right| \end{aligned}\right\} \quad (1)$$

$$\left.\begin{aligned} tile_{max x} &= \left|\frac{req_{max x}-img_{min x}}{cell_x \times block_x}\right| \\ tile_{max y} &= \left|\frac{req_{max y}-img_{min y}}{cell_y \times block_y}\right| \end{aligned}\right\} \quad (2)$$

式中，$block_x$、$block_y$ 分别表示栅格数据某一波段中切割块的长度和宽度(栅格像素数)；$tile_{min x}$、$tile_{min y}$、$tile_{max x}$、$tile_{max y}$ 分别为所请求的块号的最小列号、最小行号、最大列号、最大行号；$req_{min x}$、$req_{min y}$、$req_{max x}$、$req_{max y}$ 分别标识请求的地理范围的最小 $x$ 坐标值、最小 $y$ 坐标值、最大 $x$ 坐标值、最大 $y$ 坐标值；$img_{min x}$、$img_{min y}$ 分别表示栅格数据地理范围的最小 $x$ 坐标值、最小 $y$ 坐标值。

(5) 按照块号请求数据后，对栅格边界的块进行块内查找，通过块内切割、邻近插值后得到最终查询的栅格数据实体，本文不再赘述。

### 3.3 多源异构矢栅数据流转方法

结合矢栅数据一体化存储和检索方法，本节设计了基于空间数据引擎的流转框架[22-24]，将多源异构矢栅数据与本文提出的一体化存储模式进行转换，实现空间数据的无缝流转，如图 6 所示。

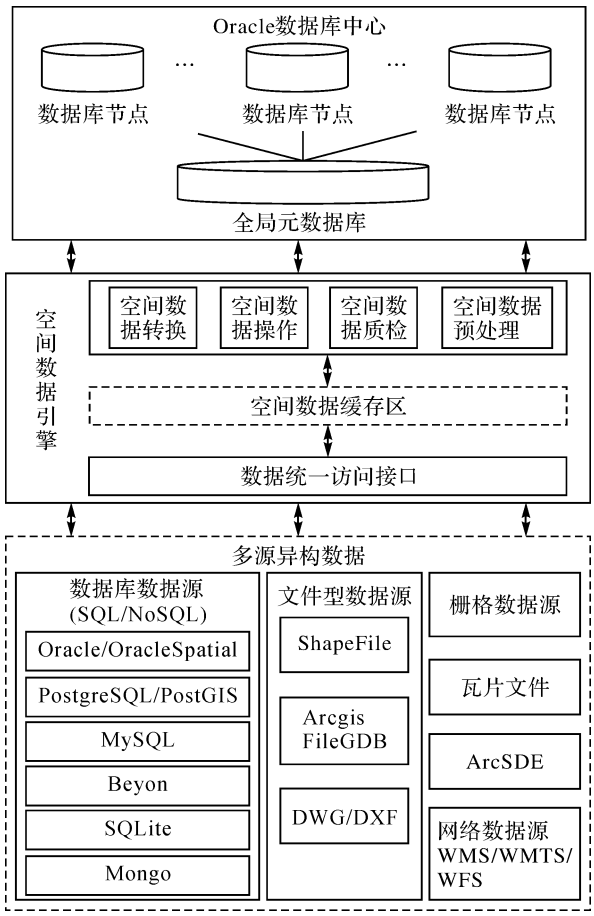



图 6 数据无缝流转框架

Fig.6 Framework of seamless data transfer

对于矢栅数据存取过程,空间数据引擎中的数据统一访问接口根据数据源地址、数据存储类型,调用相应的数据源驱动程序从局部关系数据库(Oracle、PostGIS 等)、文件型数据源(.shp、.dxf、瓦片文件等)、网络数据源(WMS、WMTS、WFS 等)中读取多源异构数据,进行空间数据一致性检查后,将处理后的数据通过数据转换接口转换为本文提出的统一数据存储模型,录入到 Oracle 中,完成数据的存取,整个过程对于用户来说是透明的。

对于矢栅数据分析过程,空间数据引擎通过数据操作接口对数据进行空间及属性过滤、空间运算、叠加分析等操作,同时提供了空间数据的预处理(空间坐标转换、数据校准等)和质检,最终将处理结果放入到数据缓存中。

完成多源异构数据互操作及处理是数据流转过程中的关键[24],在该流转框架下,本文针对关系型数据库、非关系型数据库、文件数据源、栅格数据、瓦片文件、网络数据源、第三方 SDE 等 18 种数据源开发了相应的数据驱动(Data Adapter,数据适配器),主要用于多源异构矢栅数据的存取及转换,实现将异构数据映射到统一的逻辑模型和存储结构上,达到数据的互操作和无缝流转的目的,如图 7 所示;同时为了提高数据流转的性能,数据统一访问接口采用线程池技术,通过管道将读写并行执行,降低网络传输、磁盘 I/O 带来的效率影响,并在大批量数据入库时,摒弃了传统 SQL 语句 Insert 的方式,采用数据序列化方案,生成数据文件、索引数据文件及控制脚本文件,通过 SQLLDR 工具入库,极大地提高了入库效率。

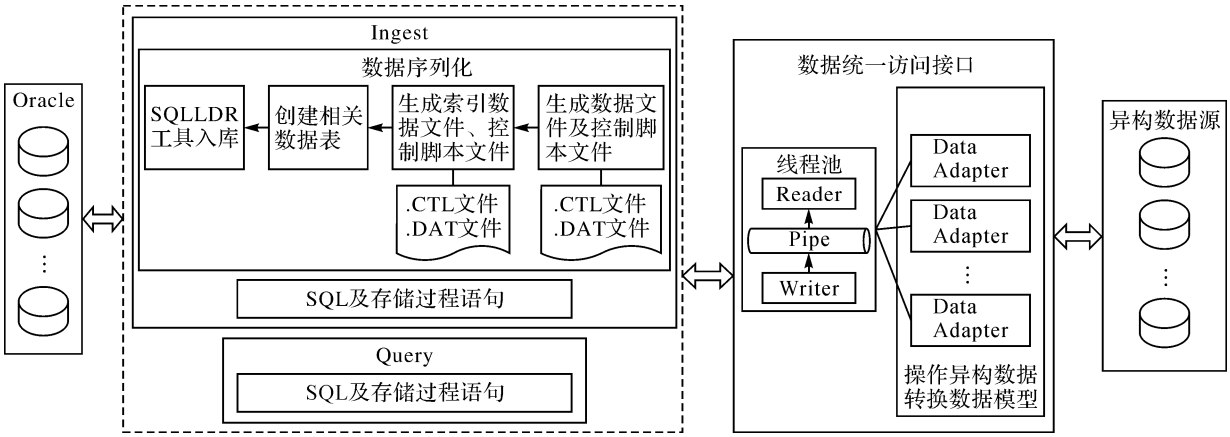



图 7　多源异构数据的互操作及处理

Fig.7　Interoperation and process of multi-source heterogeneous data

## 4　试验分析与评价

本文基于 NewMap 软件 API,利用上述方法开发了基于 Oracle 的空间数据引擎(NewMap SDE,NMSDE)。为了验证本文提出的空间数据一体化存储技术的效率和性能,与 ArcSDE[25](空间数据几何信息采用 ST_GEOMETRY 对象存储,其能够使 ArcSDE 更好地利用 Oracle 的资源、兼容 Oracle 的特性、更快地读取空间数据)在相同的计算机(CPU 8 核 3.6 GHz、4 G 内存、100 M 以太网卡)、相同的网络带宽(百兆以太网)、相同的 Oracle 数据库服务器(CPU 8 核 2.40 GHz、16 G内存、100 M 以太网卡)下进行测试。

测试数据采用 Open Street Map(OSM)提供的 2016 年 3 月某地区格式为 Shapefile 的矢量数据,数据大小 3.29 GB,共 8 个图层,其中点图层 2 个,线和面图层各 3 个;栅格数据采用的是 3 个波段 18 915 × 15 772 的 TIFF 格式影像数据 TIFF01(0.874 GB)、3 个波段 21 340 × 26 284 的 IMG 格式影像数据 IMG01(1.653 GB)和 1 个波段的 30 994 × 72 079 的 TIFF 格式的影像数据 TIFF02(2.182 GB),测试结果如下所示。

从表 1 中可以看出,NMSDE 对矢量数据的存取效率整体要优于 ArcSDE,特别是入库的数据量越大,如线数据(roads)、面数据(buildings),NMSDE 导入数据效率优势越明显,其主要原因是 NMSDE 在导入前对数据进行了序列化的预处理,减少了与数据库通信次数,降低了通信消耗时间代价。同样,对于矢量数据导出,NMSDE 采用多线程技术,并行、分批操作,效率略优于 ArcSDE。

表 1 矢量数据存取效率对比

Tab.1 Comparison of efficiency of accessing vector data to database

| 数据类型 | 图层名称 | 要素数量/万个 | NMSDE 入库耗时/s | ArcSDE 入库耗时/s | NMSDE 导出耗时/s | ArcSDE 导出耗时/s |
|---|---|---|---|---|---|---|
| 点数据 | points | 60.041 | 194.282 | 1 029.269 | 27.425 | 115.253 |
| | places | 1.952 0 | 6.825 | 33.144 | 12.777 | 4.852 |
| 线数据 | railways | 6.316 1 | 28.715 | 113.474 | 15.819 | 15.974 |
| | roads | 679.508 2 | 3 797.850 | 10 924.852 | 868.332 | 1 491.959 |
| | waterways | 32.326 6 | 221.431 | 507.663 | 66.321 | 87.726 |
| 面数据 | buildings | 346.281 3 | 1 703.978 | 5 116.084 | 238.188 | 748.724 |
| | landuse | 15.831 9 | 74.248 | 236.840 | 26.051 | 33.727 |
| | natural | 19.540 0 | 112.875 | 294.482 | 38.501 | 53.446 |

对于栅格数据的导入效率测试，本文分别采用了 128×128、256×256、512×512（ArcSDE 不支持）块大小对栅格数据进行分割导入（金字塔数据未计算在内），并且对数据未进行任何压缩。从图 8(a)中可以看出，NMSDE 的栅格数据导入效率优于 ArcSED，并且随着分块大小的递增，入库效率也有明显提升。栅格数据导出效率测试如图 8(b)所示，在相同分块大小的情况下，ArcSDE 性能略优，主要原因是在导出数据时 ArcSDE 对栅格数据进行了压缩，但是当 NMSDE 采用分块大小为 512×512 时效率最好，原因是随着栅格数据分块的递增，单次 SQL 通信过程中数据传输量增大，减少了与数据库通信次数，降低了通信代价。

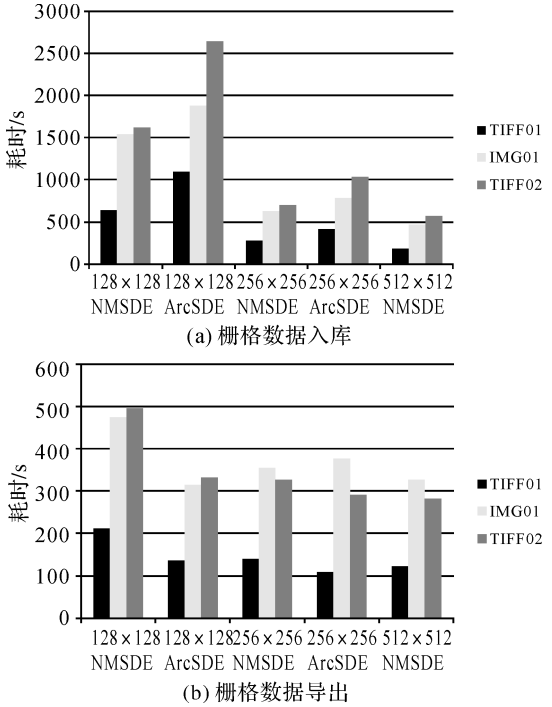



图 8 栅格数据存取效率对比

Fig.8 Comparison of efficiency of accessing raster data to database

对于空间数据的查询效率本文选取同一地理范围内的数据进行测试。如表 2 所示，对于矢量数据查询效率 NMSDE 略优于 ArcSDE，说明本文提出的优化检索技术是切实可行的；对于栅格数据查询效率如图 9 所示，NMSDE 明显优于 ArcSDE，并随着分块大小的递增效率越高，在分块大小为 512×512 时，效率最高。

表 2 矢量数据查询效率对比

Tab.2 Comparison of efficiency of querying vector data

| 数据类型 | 图层名称 | 地理范围内要素数量 | NMSDE 查询耗时/s | ArcSDE 查询耗时/s |
|---|---|---|---|---|
| 点数据 | points | 436 567 | 32.439 | 65.505 |
| | places | 12 364 | 1.257 | 1.404 |
| 线数据 | railways | 44 417 | 6.067 | 6.412 |
| | roads | 3 861 419 | 956.292 | 1 123.710 |
| | waterways | 143 583 | 59.392 | 54.031 |
| 面数据 | buildings | 2 578 889 | 466.449 | 481.855 |
| | landuse | 112 250 | 16.242 | 26.239 |
| | natural | 82 799 | 21.242 | 29.997 |

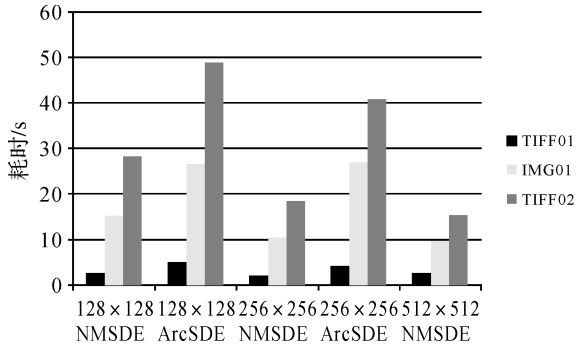



图 9 栅格数据查询效率对比

Fig.9 Comparison of efficiency of querying raster data

## 5 结束语

本文在分析当前多源、海量、异构的空间数据

存储和管理现状的基础上，提出了一种基于 Oracle 的矢栅数据一体化存储和管理方法，使用全局统一的矢栅数据存储模型及索引方法实现多源异构数据的互操作与集成管理。通过试验对比分析表明，依据本文提出的方法研发的空间数据引擎 NMSDE 在矢栅数据的存取、查询效率上优于 ArcSDE。

本文目前仅实现了 Oracle 与关系型数据库、文档存储型数据库的多源异构数据流转和互操作，随着 HBase、Accumulo 等列存储数据库以及 Hadoop MapReduce、Spark 等并行框架在 GIS 系统以及空间大数据分析中大量的应用，如何在并行分析的框架下实现基于列存储的空间数据库与关系数据库的有效集成是下一步面临的主要问题。

## 参考文献：


[1] 李德仁，姚远，邵振峰. 智慧城市中的大数据[J]. 武汉大学学报(信息科学版)，2014，39(6)：631-640.
LI Deren，YAO Yuan，SHAO Zhenfeng. Big Data in Smart City[J]. Geomatics and Information Science of Wuhan University，2014，39(6)：631-640.

[2] 李德仁. 展望大数据时代的地球空间信息学[J]. 测绘学报，2016，45(4)：379-384. DOI：10.11947/j.AGCS.2016.20160057.
LI Deren. Towards Geo-spatial Information Science in Big Data Era[J]. Acta Geodaetica et Cartographica Sinica，2016，45(4)：379-384. DOI：10.11947/j.AGCS.2016.20160057.

[3] 龚健雅. 空间数据库管理系统的概念与发展趋势[J]. 测绘科学，2001，26(3)：4-9.
GONG Jianya. Concepts and Development of Spatial Database Management[J]. Science of Surveying and Mapping，2001，26(3)：4-9.

[4] 覃雄派，王会举，李芙蓉，等. 数据管理技术的新格局[J]. 软件学报，2013，24(2)：175-197.
QIN Xiongpai，WANG Huiju，LI Furong，et al. New Landscape of Data Management Technologies[J]. Journal of Software，2013，24(2)：175-197.

[5] 胡金星，潘懋，宋扬，等. 空间数据库实现及其集成技术研究[J]. 计算机应用研究，2003，20(3)：12-14，19.
HU Jinxing，PAN Mao，SONG Yang，et al. Research on Realization and Integrated Technology of Spatial Databases[J]. Application Research of Computers，2003，20(3)：12-14，19.

[6] 陈传彬，邬群勇，陈崇成. 基于 GML 的多源异构空间数据集成框架研究[J]. 测绘科学，2005，30(5)：53-55.
CHEN Chuanbin，WU Qunyong，CHEN Chongcheng. A GML-based Multi-source and Heterogeneous Spatial Data Integration Framework[J]. Science of Surveying and Mapping，2005，30(5)：53-55.

[7] Oracle Corporation. Oracle® Spatial Developer's Guide 11 g Release 2(11.2)[EB/OL]. (2013-11). http://docs.oracle.com/cd/E11882_01/appdev.112/e11830/toc.htm.

[8] Oracle Corporation.Oracle® Spatial GeoRaster Developer's Guide 11 g Release 2(11.2)[EB/OL]. (2013-11). http://docs. oracle. com/cd/E11882 _ 01/appdev. 112/e11827/toc.htm.

[9] 申德荣，于戈，王习特，等. 支持大数据管理的 NoSQL 系统研究综述[J]. 软件学报，2013，24(8)：1786-1803.
SHEN Derong，YU Ge，WANG Xite，et al. Survey on NoSQL for Management of Big Data[J]. Journal of Software，2013，24(8)：1786-1803.

[10] 陈崇成，林剑峰，吴小竹，等. 基于 NoSQL 的海量空间数据云存储与服务方法[J]. 地球信息科学学报，2013，15(2)：166-174.
CHEN Chongcheng，LIN Jianfeng，WU Xiaozhu，et al. Massive Geo-spatial Data Cloud Storage and Services Based on NoSQL Database Technique[J]. Journal of Geo-information Science，2013，15(2)：166-174.

[11] YAN Baoqiang，RHODES P J. IDEA-An API for Parallel Computing with Large Spatial Datasets[C]// Proceedings of the 2011 International Conference on Parallel Processing. Taipei，China：IEEE，2011：355-364.

[12] 康俊锋. 云计算环境下高分辨率遥感影像存储与高效管理技术研究[D]. 杭州：浙江大学，2011.
KANG Junfeng. Technologies of Storage and Efficient Management on Cloud Computing for High Resolution Remote Sensing Image[D]. Hangzhou：Zhejiang University，2011.

[13] 雷德龙，郭殿升，陈崇成，等. 基于 MongoDB 的矢量空间数据云存储与处理系统[J]. 地球信息科学学报，2014，16(4)：507-516.
LEI Delong，GUO Diansheng，CHEN Chongcheng，et al. Vector Spatial Data Cloud Storage and Processing Based on MongoDB[J]. Journal of Geo-information Science，2014，16(4)：507-516.

[14] 邬伦，张毅. 分布式多空间数据库系统的集成技术[J]. 地理学与国土研究，2002，18(1)：6-10.
WU Lun，ZHANG Yi. The Integrated Framework on Distributed Multi-Spatial Database System[J]. Geography and Territorial Research，2002，18(1)：6-10.

[15] 刘小飞，关昆，于海波，等. 多源多目标空间数据库的一体化集成与管理技术研究[J]. 测绘通报，2014(12)：97-100.
LIU Xiaofei，GUAN Kun，YU Haibo，et al. Research of Integration and Management of Multi-source Multi-target Geospatial Database[J]. Bulletin of Surveying and Mapping，2014(12)：97-100.

[16] 徐道柱，罗斌，周岩，等. 分布式环境下矢栅一体化数据组织模型设计[J]. 地球信息科学学报，2016，18(12)：1588-1596.

XU Daozhu, LUO Bin, ZHOU Yan, et al. Model Design of Integrated Vector and Raster Data Organization under the Distributed Environment[J]. Journal of Geo-information Science, 2016, 18(12): 1588-1596.

[17] 王意洁, 孙伟东, 周松, 等. 云计算环境下的分布存储关键技术[J]. 软件学报, 2012, 23(4): 962-986.
WANG Yijie, SUN Weidong, ZHOU Song, et al. Key Technologies of Distributed Storage for Cloud Computing [J]. Journal of Software, 2012, 23(4): 962-986.

[18] Ozsu M T, Valduriez P. Principles of Distributed Database Systems[M]. 2nd ed. New York: Prentice Hall, 1999.

[19] 李德仁, 关泽群. 空间信息系统的集成与实现[M]. 武汉: 武汉测绘科技大学出版社, 2000.
LI Deren, GUAN Zequn. Integration and Implementation of Spatial Information System[M]. Wuhan: Wuhan Surveying and Mapping Science and Technology University Press, 2000.

[20] KJENSTAD K.On the Integration of Object-based Models and Field-based Models in GIS[J]. International Journal of Geographical Information Science, 2006, 20(5): 491-509.

[21] GOODCHILD M F, YUAN M, COVA T J. Towards a General Theory of Geographic Representation in GIS[J]. International Journal of Geographical Information Science, 2007, 21(3): 239-260.

[22] 朱欣焰, 张建超, 李德仁, 等. 无缝空间数据库的概念、实现与问题研究[J]. 武汉大学学报(信息科学版), 2002, 27(4): 382-386.
ZHU Xinyan, ZHANG Jianchao, LI Deren, et al. Concepts, Implementation and Problems of the Seamless Spatial Database[J]. Geomatics and Information Science of Wuhan University, 2002, 27(4): 382-386.

[23] LIM E P, CHIANG R H L. The Integration of Relationship Instances from Heterogeneous Databases [J]. Decision Support Systems, 2000, 29(2): 153-167.

[24] 金宝轩. 网格环境下的异构空间数据库集成技术[J]. 计算机工程, 2008, 34(5): 74-76.
JIN Baoxuan. Integration of Heterogeneous Spatial Databases in Grid Environment[J]. Computer Engineering, 2008, 34(5): 74-76.

[25] 杜红悦, 刘先林, 宫辉力. 基于 ArcSDE 的空间数据综合管理系统设计与实现[J]. 测绘科学, 2009, 34(5): 171-173.
DU Hongyue, LIU Xianlin, GONG Huili. Design and Implementation of Spatial Data Integreted Management System Based on ArcSDE[J]. Science of Surveying and Mapping, 2009, 34(5): 171-173.

(责任编辑:宋启凡)

**收稿日期**: 2016-11-17
**修回日期**: 2017-03-27

**第一作者简介**: 吴政(1982—),男,博士,助理研究员,主要从事空间数据组织与管理、地理信息系统建模及地理信息服务等研究。
**First author**: WU Zheng(1982—), male, PhD, assistant researcher, majors in spatial data organization and management, GIS modeling and geographic information service.
**E-mail**: wuzheng_gucas@163.com
**通信作者**: 李成名
**Corresponding author**: LI Chengming
**E-mail**: cmli@casm.ac.cn