维普资讯 http://www.cqvip.com

基于区间编码的GML索引与查询*)

朱付保[1] 关佶红[1,2] 周水庚[3]
(武汉大学计算机学院 武汉 430079)[1] (同济大学计算机科学与技术系 上海 201804)[2]
(复旦大学计算机科学与工程系 上海 200433)[3]

摘 要 GML很好地解决了不同GIS系统间地理数据的共享、交换和集成问题,已逐步成为空间数据编码、传输、存储和发布的国际标准。随着GML的广泛应用,如何有效地管理GML数据已成为亟待解决的问题。本文提出一种基于区间编码的方法对GML文档中的元素、属性、文本、几何体等要素进行编码,对非空间特性信息建立B树以实现值查询,对空间特性信息建立R树索引或四叉树索引以进行空间分析。在查询处理中,采用以XQuery为基础的GQL查询语言,以实现对GML文档的非空间查询、空间查询及混合查询。实验证明这种方法能够有效地处理在GML文档上进行的值查询和空间分析操作。

关键词 地理标记语言,区间编码,索引,值查询,空间分析

Interval Coding based GML Indexing and Querying

ZHU Fu-Bao[1] GUAN Ji-Hong[1,2] ZHOU Shui-Geng[3]
(School of Computer, Wuhan University, Wuhan 430079)[1]
(Department of Computer Science and Technology, Tongji University, Shanghai 201804)[2]
(Department of Computer Science and Engineering, Fudan University, Shanghai 200433)[3]

Abstract Geography Markup Language is now becoming the de facto standard developed by OGC to standardize the representation of geographical data in XML, which makes the sharing, exchanging and integration of data easier in different geographical information system. More and more geographical data are presented in GML format with its wide application, and thus arouse the problem of how to efficiently realize retrieval and spatial analysis in GML document. This paper proposes an interval coding method to encode the element, attribute, text, and geometry information in GML, to build B tree for value query and R tree or quad tree for spatial analysis. Extended XQuery is used for GML non-spatial, spatial and hybrid query. Experiment shows that this method is effective and efficient.

Keywords Geography markup language, Interval coding, Index, Query, Spatial analysis

1 引言

地理标记语言GML[1]是开放地理信息系统组织OGC为了解决WebGIS环境下不同格式地理数据共享而制定的一套基于XML的地理信息编码工具。目前GML作为一种可扩展的、标准化的地理信息编码方式,为地理信息的建模、传输、存储和发布提供了一个公共的地理对象描述标准,从而使得各个独立开发的应用之间的互操作成为可能。GML也由原来的三个基模式(Base Schema)的2.x版发展为由30个基模式组成的3.x版。GML描述地理信息的能力越来越强,不仅解决了空间数据格式不一致的问题,而且提供包含结构和语义的数据表达,符合当前语义Web的要求,使地理信息在不同系统间的交换、集成和共享变得更加容易。

随着GML在诸多领域的广泛应用,越来越多的地理信息以GML格式来描述。由于GML文档相当大,以文件方式来管理地理空间数据很难提供较好的空间信息查询、空间数据分析、存取控制、并发控制等功能。如何有效地管理GML数据已成为亟待解决的问题。Corcoles等[2]分析比较了基于RDBMS的3个存储模型LegoDB,Monet,Xparent用于存储GML数据时可扩展性及查询时间;Sripada等[3]列举了GML领域的研究问题,如GML存储、解析、查询、可视化和GML在移动设备上的应用等。Zhu等[4]利用支持空间数据操作的对象关系数据库系统对GML文档进行管理。对于GML的查询处理,Corcoles等[5]提出了一种基于XML-QL的GML查询语言规范,该规范基于的数据模型和代数支持空间特征。Vatsavai[6]比较了几种XML查询语言,提出了基于XQuery的GML查询语言GML-QL,描述了GML-QL的结构,给出了具体的GML文档查询实例,但没有给出相应的数据模型、语法结构和语义处理。Guan等[7]在XQuery的基础上提出了GML查询语言GQL,定义了空间数据类型,增加了空间处理算子并给出了相应的形式语义。

为了提高GML查询效率,必须在GML数据的基础上,建立相应的索引。半结构化数据索引技术近年来也得到广泛深入的研究,按索引模式分为三类:路径索引、编码-连接索引[8~10]和序列匹配索引。路径索引提取XML文档中的全部或部分路径信息建立索引,以便在查询的时候减少查询数据

*)基金项目:本课题得到国家自然科学基金(60573183)、新世纪优秀人才支持计划(NCET-06-0376)、测绘遥感信息工程国家重点实验室开放基金(WKL(04)0303)资助。朱付保 博士研究生,主要研究方向:空间数据库,数据挖掘,人工智能。

维普资讯 http://www.cqvip.com

空间，提高查询效率。但对于结构复杂、规模大的文档，路径索引的规模也会变得很大。编码-连接索引将 XML 文档树中的结点按照一定的方法进行编码，通过比较结点之间的编码，快速地确定结点之间的结构关系（父/子关系，祖先/后代关系），使用起来简单灵活，容易应用到关系数据库系统中。序列匹配索引采用编码方法将 XML 文档数据和 XML 查询表达式编码成序列，使用编码序列代替路径作为查询的基本单元，将 XML 路径表达式查询等价成发现查询表达式编码序列在文档数据编码序列中的子模式匹配过程。

将 GML 文档存储到支持空间数据操作的对象关系数据库中，GML 非空间数据作为普通字段，空间数据作为一个对象，利用对象关系数据库成熟的技术管理 GML 数据是一种可行的解决方案。为了提高 GML 的查询效率，本文提出了一种基于区间编码的索引方法，对文档中的元素、属性、文本和集合体等要素进行编码，对文档中非空间数据建立 B^+-树索引，对空间数据建立 R 树索引或四叉树索引，以满足值查询和空间分析操作。

2 GML 文档的区间编码方案与索引结构

与 XML 一样，GML 文档也可以看作一个树形结构，区间编码方案为树中每一个结点赋予一个区间编码$[C_1, C_2]$，通过结点的区间编码确定结点间的父子关系和子孙关系。

2.1 编码方案

Dietz 编码[8]要扫描两遍文档才能为每个结点进行编码，Li 编码[9]不易确定结点间的兄弟关系和前驱后继关系。考虑到 GML 文档往往很大，应尽量减少编码所花时间，在最少的扫描遍数里，对每个几点进行编码。为此，我们扩展了 Zhang 编码[10]，增加结点的父结点的序号和结点类别，只需一遍扫描，即可将 GML 文档树中每个结点表示为一个五元组：

$(docID, firstOrder, lastOrder, parentOrder, kind)$

其中，$docID$ 是文档的编号，$firstOrder$ 是前序遍历文档树期间首次被访问时的序号，$lastOrder$ 是最后一次被访问时的序号。$parentOrder$ 是该结点父结点的 $firstOrder$，$kind$ 是该结点的类别（如元素结点、属性结点、值结点或几何体结点等）。图 1 给出了文档树的六元组的中间三元表示。

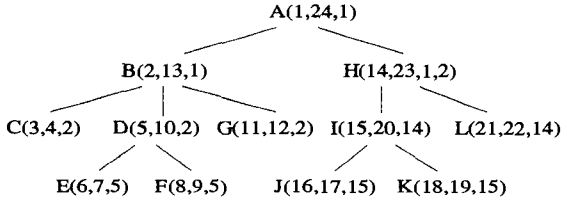

图 1 基于区间的 GML 文档树编码

根据上述编码方案，容易证明如下结论：文档树中的任意结点 u，其祖先结点 v 的集合为：

$$\{v \mid firstOrder(v) < firstOrder(u) \ and \ lastOrder(v) > firstOrder(u)\} \quad (1)$$

其双亲结点 v 满足：

$$\{v \mid firstOrder(v) = parentOrder(u)\} \quad (2)$$

其子孙结点 v 的集合为：

$$\{v \mid firstOrder(v) > firstOrder(u) and \ firstOrder(v) < lastOrder(u)\} \quad (3)$$

其孩子结点 v 的集合为：

$$\{v \mid parentOrder(v) = firstOrder(u)\} \quad (4)$$

其兄弟结点 v 的集合为：

$$\{v \mid parentOrder(v) = parentOrder(u)\} \quad (5)$$

2.2 索引与数据组织

GML 数据支持值查询和结构查询，值查询可以通过 GML 文档名、元素名/值，属性名/值进行；结构查询可以通过查询表达式中祖先/子孙关系进行。

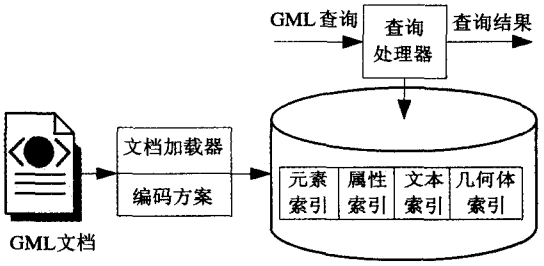

图 2 GML 数据索引结构

基于区间编码的 GML 索引结构由元素索引、属性索引、文本索引及几何体索引组成，如图 2 所示。对于元素、属性和文本结点，以 B^+-树方式组织索引，将结点的名字的标识符作为关键字，叶子节点指向具有同名的元素的一组定长记录，如图 3 所示，每个索引项记录了结点的区间编码及类型（元素、属性或值结点）。

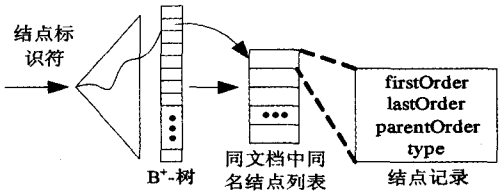

图 3 结点索引结构

对于几何体结点，以 R-树或四叉树方式组织索引，如图 4 所示，索引项记录了几何体结点的区间按编码及其形状（点、线或多边形）。R-树是一棵平衡的多路查找树，所有的叶结点都在同一层，并存储几何体对象的最小接矩形 MBR（Minimum Bounding Rectangles）。

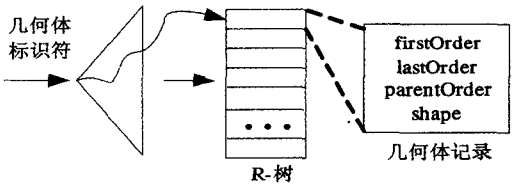

图 4 几何体索引结构

3 GML 查询处理

对 GML 文档进行编码后，对 GML 文档的查询就转变为在编码空间上的查询。本节详细讨论 GML 文档的查询处理过程，并根据一个 GML 文档实例，说明基于区间编码的索引在查询中的应用。文档实例表示一个包含道路和地区的地图，图 5 给出了该文档部分数据。

3.1 GML 查询语言

本文采用 GQL[7]作为 GML 的查询语言，GQL 是在 XQuery 的基础上，增加了空间数据类型、空间操作算子和形式语义，支持非空间数值查询、空间分析与运算、混合查询。

维普资讯 http://www.cqvip.com

表1给出了三个GML查询实例，其中，Q1用于查询所有类型为“Multi-lane”的公路，Q2查询每条公路所经过的地区，Q3查询类型为“Multi-lane”的公路所经过的地区。

```
<Map gml:id="M106">
 <boundedBy><gml:Box><gml:coord>
  <gml:X>-158.20</gml:X><gml:Y>19.06</gml:Y></gml:coord>
  <gml:coord><gml:X>-67.097</gml:X><gml:Y>70.32</gml:Y>
 </gml:coord></gml:Box></boundedBy>
 <FeatureMember>
  <Road id="Rd01">
   <admin>Interstate</admin>
   <type>Multi-Lane</type>
   <centerLineOf><gml:LineString><gml:coord><gml:X>-76.16</gml:X>
    <gml:Y>41.05</gml:Y></gml:coord><gml:coord>
    <gml:X>-77.83</gml:X><gml:Y>40.98</gml:Y></gml:coord> ...
    <gml:coord><gml:X>-80.16</gml:X><gml:Y>41.19</gml:Y>
    </gml:coord></gml:LineString>
   </centerLineOf >
  </Road> ...
  <Road id="Rdxx">    ...    </Road>

  <District id="Dt01"><name>New York</name>
   <pop>18195566</pop></SimpleProperty>
   <extentOf><gml:Polygon><gml:coord><gml:X>-79.76</gml:X>
    <gml:Y>42.27</gml:Y></gml:coord><gml:coord>
    <gml:X>-76.91</gml:X><gml:Y>43.28</gml:Y></gml:coord> ...
    <gml:coord> <gml:X>-79.76</gml:X><gml:Y>42.00</gml:Y>
    </gml:coord></gml:Polygon></extentOf>
  </District> ...
  <District gml:id="Dtxx">    ...    </District>
 </FeatureMember>
</Map>
```

图5　GML文档实例片断

表1　GML查询实例

编号	查询表达式	查询类型
Q1	FOR $r IN document(“map. xml”) //Road WHERE $r/type = “Multi-Lane” RETURN $r	非空间查询

Q2	FOR $r IN document(“map. xml”)//Road,
	$d IN document(“map. xml”)//District
	WHERE Cross($r/centerLineOf, $d/extentOf)=1
	RETURN
	RiverDistricts〉
	〈Rid〉$r/@gml:id〈/Rid〉
	〈Dname〉$d/name〈/Dname〉
	〈/RiverDistricts〉
		空间相交查询

Q3	FOR $r IN document(“map. xml”)//Road,
	$d IN document(“map. xml”)//District
	WHERE Cross($r/centerLineOf, $d/extentOf)=1
	AND $r/type =“Multi-Lane”
	RETURN
	〈RiverDistricts〉
	〈Rid〉$r/@gml:id〈/Rid
	〈Dname〉$d/name〈/Dname〉
	〈/RiverDistricts〉
		混合查询

3.2　查询分析与处理

下面以混合查询Q3为例，来分析查询的执行过程。在这个查询表达式中，待查询的地理数据以GML格式存放在“map. xml”文档中，该文档中存放着道路(Road)和地区(District)的非空间信息(道路的管辖、使用类别，地区的名字、人口)及空间信息(道路的中心线，地区的覆盖范围)。查询表达式Q3里包含了查询的非空间条件，即元素type的值必须为“Multi-Lane”，及空间条件即Road和District必须相交，查询结果仍以GML格式来表示。该查询表达式的处理步骤如下：

S1. 查询分解：将WHERE子句中查询条件分解为一个非空间查询($r/type = “Multi-Lane”)和一个空间查询(Cross($r/centerLineOf, $d/extentOf)=1)，即这个查询是Q1查询和Q2查询的交集；

S2. 查询优化：一般地，同时存在非空间查询和空间查询条件时，为了提高查询效率，先处理非空间查询，在此基础上，再进行空间分析与运算；

S3. 查询执行：对于非空间查询，根据结点的B^+-树索引结构，快速定位满足条件的元素和文本结点；在此基础上，利用R-树索引结构，进行空间条件计算，找到满足条件的集合体结点；

S4. 结果输出：对多个查询条件的查询结果进行合并，按照查询表达式中RETURN语句指定的输出格式，将查询和计算结果以GML格式输出。

4　实验及分析

为测试基于区间编码的GML文档索引与查询性能，我们用Java语言实现了一个原型系统、主要开发软件和开发工具，包括DOM4J、SAX、JGeometry、Oracle9i、Oracle Spatial和JBuilder2006等。实验运行在Windows XP Professional环境下，硬件配置为AMD Athlon 64 Processor 1.79GHz处理器、1GB内存、120GB硬盘空间。系统中的地理数据取自ArcView GIS3.2和MapInfo6.0中的实例。

4.1　数据集与性能分析

为了便于测试性能，我们对六组数据进行了测试，文档大小从0.5MB到20MB不等，并对其中的总结点数和几何体结点树进行了统计，如表2所示。

表2　不同大小文档及包含结点数

文档大小(MB)	结点总数(个)	几何体结点数(个)
0.5	2134	126
1	5007	295
2	10736	632
5	259021	15237
10	540065	31769
20	1182903	69583

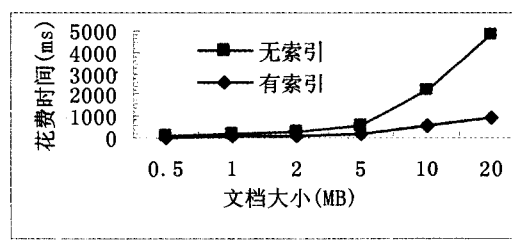

图6　非空间查询Q1在有无索引时花费时间对比

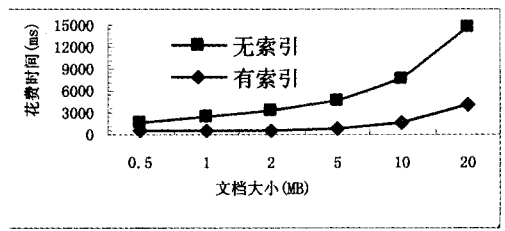

图7　空间查询Q2在有无索引时花费时间对比

(下转第83页)

维普资讯 http://www.cqvip.com

(略逊于 MON-Tree,但优于 FNR-Tree)。

将来的研究工作包括对 UTR-Tree 做进一步的优化,并将在 Net-MOD 原型系统的基础上进行更为详细的实验比较与分析。此外,我们还将设计完整的不确定性数据类型和查询操作,并给出相应操作在 UTR-Tree 支持下的实现算法。

参 考 文 献

1 Pfoser D, Jensen C S. Capturing the Uncertainty of Moving Object Representations. In:Proc. of SSD'99, Hong Kong, China, July 1999

2 Trajcevski G, Wolfson O, Chamberlain S, et al. The Geometry of Uncertainty in Moving Objects Databases. In: Proc. of EDBT'02, Prague, Czech Republic, 2002

3 Pfoser D, Tryfona N. Capturing Fuzziness and Uncertainty of Spatiotemporal Objects. In: ADBIS 2001, Vilnius, Lithuania, Sept. 2001

4 Trajcevski G, Wolfson O, Cao H, et al. Managing Uncertain Trajectories of Moving Objects with Domino. In: ICEIS'02, Spain, April 2002

5 T∅ssebro E, Nygärd M. Uncertainty in Spatiotemporal Databases. In:Proc. 2nd Biennial Int Conf. on Advances in Information Systems (ADVIS), Turkey, Oct. 2002

6 Ding Z, Güting R H. Uncertainty Management for Network Constrained Moving Objects. In: Proc. of DEXA'2004, Zaragoza, Spain, August 2004

7 Gowrisankar H, Nitte S. Reducing Uncertainty in Location Prediction of Moving Objects in Road Networks. In: Proc. of GIScience 2002,Colorado, 2002

8 Meratnia N, Kainz W, et al. Spatio-temporal Methods to Reduce Data Uncertainty in Restricted Movement on a Road Network. In: Proc. of joint ISPRS, IGU and CIG Symp on Geospatial theory, Processing and Applications, Canada, 2002

9 Almeida V T, Güting R H. Supporting Uncertainty in Moving Objects in Network Databases. In: Proc. of the 13th Intl Workshop on Geographic Information Systems (ACM-GIS), Bremen, Germany, 2005

10 Pfoser D, Jensen C S, Theodoridis Y. Novel Approach to the Indexing of Moving Object Trajectories. In: Proc. of the 26th VLDB,Cairo, Egypt, 2000

11 Saltenis S, Jensen C S, Leutenegger S T, et al. Indexing the Position of Continuously Moving Objects. In: Proc. of ACM SIGMOD 2000, TX, USA, 2000

12 Frentzos E. Indexing objects moving on fixed networks. In: Proc the 8th Int Symp Spatial and Temporal Databases, Santorini Island, Greece, 2003

13 Almeida V T, Güting R H. Indexing the Trajectories of Moving Objects in Networks. GeoInformatica,2005,9(1)

14 Chen J*, Meng X. Indexing Future Trajectories of Moving Objects in a Constrained Network. Journal of Computer Science and Technology,2007,22(2): 245～251

15 Ding Z, Güting R H. Managing Moving Objects on Dynamic Transportation Networks. In: Proc. of the 16th International Conference on Science and Statistical Database Management (SSDBM'2004), Santorini, Greece, June 2004

(上接第 67 页)

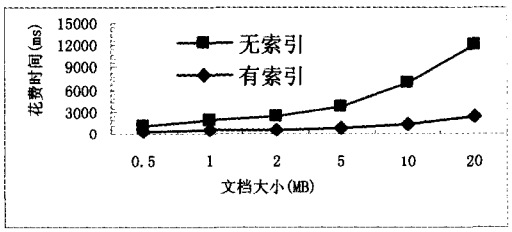

图 8 混合查询 Q3 在有无索引时花费时间对比

文档中地理数据涵盖了道路、河流、湖泊、城市、省区、水域、排水等,涉及到的几何对象包括 Point、LineString、LinearRing、Box、Polygon、MultiLine、MultiPolygon 等。图 6、图 7 和图 8 分别给出了这三种查询在无索引和有索引时所花费的时间对比。

从图 6～8 可以看出,非空间查询 Q1 由于不涉及空间运算,查询耗时最短,与无索引时相比,建立 B^+-树索引后,查询效率有一定程度的提高;空间查询 Q2 在建立 R 树索引后,查询效率有显著提高;混合查询 Q3 先经过非空间条件过滤后,查询时间明显比 Q2 有所下降,查询效率也比无索引时有显著提高。

结束语 本文提出一种基于区间编码的 GML 文档索引与查询方法,将 GML 文档树中结点按前序遍历中第一次和最后一次被访问时的次序进行编码。对元素、属性、文本结点以 B^+-树方式来组织索引,以提高值查询和结构查询的查询速度;对几何体结点按 R-树方式组织索引,以便提高空间查询和分析效率。查询语言采用基于 XQuery 的 GML 查询语言 GQL,并对三种查询(非空间查询、空间查询及混合查询)进行了分析与比较。实验证明,本文所提出的 GML 文档编码方案和索引机制是可行的,能够有效地处理在 GML 文档上进行的值查询和空间分析操作。

参 考 文 献

1 Cox S, Daisey P, Lake R, Portele C, Whiteside A. Geography Markup Language (GML) Implementation Specification version 3.0 [S], OpenGIS Consortium, 2003

2 Córcoles J E, González P. Analysis of Different Approaches for Storing GML Documents [C]. In: Proceedings of ACM GIS'02, 2002. 11～16

3 Sripada L N, Lu C, Wu W. Evaluating GML Support for Spatial Databases. In:COMPSAC Workshops, 2004. 74～77

4 Zhu F, Guan J, Zhou J, Zhou S. Storing and Querying GML in Object-Relational Databases [C]. In: Proceedings of ACM GIS'06,2006. 107～114

5 Corcoles J E, Gonzalez P. A Specification of a Spatial Query Language over GML [C]. In: Proceedings of ACM GIS'' 01, 2001. 112～117

6 Vatsavai R R. GML-QL: A Spatial Query Language Specification for GML. http://www. cobblestoneconcepts. com/ucgis2summer2002/vatsavai /vatsavai. htm, 2002

7 Guan J, Zhu F, Zhou J, Niu L. GQL: Extending XQuery to query GML documents [J]. Geo-Spatial Information Science, 2006,9(2):118～126

8 Dietz P F. Maintaining Order in a Linked List [C]. In: Proceedings of the Annual ACM Symposium on Theory of Computing, 1982. 122～127

9 Li Q, Moon B. Indexing and Querying XML Data for Regular Path Expressions [C]. In: Proceedings of VLDB'01, 2001. 361～370

10 Zhang C, Naughton J, DeWitt D, Luo Q, Lohman G. On Supporting Containment Queries in Relational Database Management Systems [C]. In: Proceedings of SIGMOD'01, 2001. 426～437