

# 一种面向 WFS 服务的分布式空间连接查询优化策略

蓝贵文<sup>1</sup> 黄全义<sup>2</sup> 周晓青<sup>3</sup>

(1 武汉大学测绘学院,武汉市珞喻路 129 号,430079)  
(2 清华大学公共安全研究中心,北京市海淀区,100084)  
(3 中国测绘科学研究院地理信息工程国家测绘局重点实验室,北京市北太平路 16 号,100039)

**摘要:**研究了一般分布式环境下空间连接查询的优化策略,针对 OGC 的 WFS 服务规范的特点,提出了一种基于递归区域划分的 WFS 服务空间连接查询优化策略。实验结果表明,当查询区域范围内包含的数据量越大时,该策略越能有效地降低查询需要的网络数据传输费用,缩短查询执行时间。  
**关键词:**网络要素服务;空间连接;查询优化;递归区域划分  
**中图法分类号:**P208

随着空间信息应用规模的扩大,研究如何从分布式的空间数据库中查询符合条件的空间对象集合,具有重要的应用价值。开放地理信息联盟(OGC)的网络要素服务(WFS)规范定义了基于 HTTP 协议访问和更新地理要素的接口,在 ArcIMS 等多个 GIS 软件产品中实现。可以预见,WFS 将成为网络环境下重要的空间数据源之一。对于查询 1 而言,假定消防站和商场的数据分别通过两个 WFS 提供,分布式空间连接(distributed spatial join, DSJ)策略就是直接从两个远端的 WFS 服务器把空间对象数据下载到本地,再作连接运算。在参与空间连接的空间对象较大的情况下,这种方法导致数据传输费用较高。另外,由于 WFS 以 GML 格式传输空间数据,一般情况下,WFS 服务器需要把空间数据从其他格式转换为 GML 文本流,而执行空间连接运算的站点需要对 GML 文本流进行解析,所需的计算量非常大。因此,本文主要研究如何改善执行策略,剪除不参与最后连接操作的空间元组,从而减少数据传输费用,同时减少 GML 数据的生成和解析所需的计算量,缩短查询执行时间。

## 1 研究思路和方法

目前,空间半连接(spatial Semijoin, SSJ)策

略<sup>[1-3]</sup>被认为是减少网络传输数据量最有效的解决方法之一。SSJ 策略需要在参与连接操作的空间数据集有明显的数据倾斜,才能有效地降低网络传输的数据量。基于区域递归划分<sup>[4]</sup>(partition based spatial-merge, PBSM)的策略是通过对整体上没有表现出数据倾斜的两个数据集进行区域划分,利用局部区域表现出的数据倾斜特性来达到降低网络数据传输费用的目的。

### 1.1 空间半连接查询

定义 1<sup>[1]</sup> 设  $R$  和  $S$  为两个关系,  $A$  和  $B$  分别是  $R$  和  $S$  上的空间属性,则  $R$  和  $S$  关于空间关系  $\theta$  的空间半连接为  $R' = \{r \in R \mid \exists s \in S, \text{使得 } f_R(r, A) \varphi f_S(s, B)\}$ , 记为  $R' = R \bowtie_{A \theta B}^S S$ 。其中,  $f_R$  和  $f_S$  为空间几何近似映射函数,用于将复杂的空间对象映射为简单的几何对象(如最小包围矩形 MBR);  $\theta$  和  $\varphi$  为空间连接谓词,  $\varphi$  依赖于  $\theta$ 。

设  $R$  和  $S$  分别存储于站点  $R_{\text{site}}$  和  $S_{\text{site}}$  上,要求求在  $S_{\text{site}}$  上得到空间连接结果  $R \bowtie_{A \theta B}^S S$ , 则从  $S_{\text{site}}$  发起的空间半连接查询过程如下<sup>[1,3]</sup>: ① 令  $B' = \{s, B \mid s \in S\}$ ; ② 令  $B_f = \{f_S(b) \mid b \in B'\}$ ; ③ 将  $B_f$  从  $S_{\text{site}}$  传到  $R_{\text{site}}$ ; ④ 在  $R_{\text{site}}$  计算空间半连接  $R'$ , 即  $R' = \{r \in R \mid \exists b \in B_f, \text{使得 } f_R(r, A) \varphi f_S(s, B)\}$ , 其中  $\varphi = g(\theta)$ ; ⑤ 将  $R'$  从  $R_{\text{site}}$  传到  $S_{\text{site}}$ , 在  $S_{\text{site}}$  上

执行空间连接  $R' \overset{S}{\bowtie}_{A \theta B} S$ 。

文献[1]已证明  $R' \overset{S}{\bowtie}_{A \theta B} S$  等价于  $R \overset{S}{\bowtie}_{A \theta B} S$ 。设  $R$ 、 $S$  数据集每个元组的平均大小为  $r^0$ 、 $s^0$ ，取  $f_s$  为求取空间对象的最小包围矩形，数据大小为  $mbr^0$ ，则传输的数据量为：

$$C_{SSJ} = mbr^0 \cdot |S| + r^0 \cdot |R'| \quad (1)$$

如果直接把  $R$  送到  $S_{site}$  (DSJ 策略)，则产生的网络传输费用为  $r^0 \cdot |R|$ 。显然，当

$$mbr^0 \cdot |S| < r^0 \cdot (|R| - |R'|) \quad (2)$$

成立时，SSJ 策略才会起到优化效果，否则应考虑采用 DSJ 策略。本文用被过滤掉的空间对象个数与不作优化时所需传输的空间对象个数的比值来表征空间半连接的过滤作用，称之为过滤率。在式(2)中，过滤率为  $(|R| - |R'|)/|R|$ 。过滤率越大，则 SSJ 策略的优化效果越明显。

$|R'|$  一般事先较难精确获知，其估算方法取决于空间拓扑关系、距离关系、方向关系等，在不同条件下，其差异较大。假设  $R$  呈均匀分布，则  $|R'|$  的估计值为：

$$|R'| = \text{area}(\bigcup_{s \in S} s, \text{MBR}) \cdot |R| / \text{area}(D) \quad (3)$$

式中，area 为求面积的函数； $D$  为连接区域。

### 1.2 基于递归区域划分的分布式查询策略

基于递归区域划分的分布式查询方法<sup>[3,4]</sup>是通过最小化每一区域划分的网络数据的传输费用来获取整个查询区域的网络数据传输费用的最小化，提高对于数据倾斜的敏感程度。

定义 2<sup>[3]</sup> 设连接区域为  $D$ ， $\pi(D)$  为  $D$  的一个空间子集族， $\pi(D) = \{D_i | i=1, \dots, N, D_i \subseteq D \wedge D_i \neq \emptyset\}$ ，且满足：①  $\bigcup_{i=1}^N D_i = D$ ；②  $i \neq j \Rightarrow D_i \cap D_j \neq \emptyset$ ，则称  $\pi(D)$  为  $D$  的一个划分， $D_i$  为划分  $\pi(D)$  的一个块。

若  $R_i$  和  $S_i$  分别为  $D_i$  对应的元组集合，在二维空间中，如果空间连接谓词仅包含拓扑谓词时，则有：

$$R \overset{S}{\bowtie}_{A \theta B} S = \bigcup_{i=1}^N R_i \overset{S_i}{\bowtie}_{A \theta B} S_i \quad (4)$$

即连接任务  $R \overset{S}{\bowtie}_{A \theta B} S$  可以在该区域划分下由多个子任务  $R_i \overset{S_i}{\bowtie}_{A \theta B} S_i$  来完成，并剔除子任务结果中包含的重复匹配记录。如图 1 所示， $R_i$ 、 $S_i$  ( $i=1, 2, 3, 4$ ) 为两个数据集中对应的区域分块。从图 1 (a) 可以看出，处于分区边缘上的空间对象被分到两个子区域中，造成计算结果可能产生重复的匹

配记录。文献[4]研究了分布式空间数据库情形下如何把边缘上的空间对象分配到各子区域的方法，但是在面向 WFS 服务条件下进行区域划分时，产生重复是不可避免的。

对于每个分块  $D_i$ ，首先评估该分块可能导致的网络传输费用，然后决定该分块的执行策略。如果满足  $mbr^0 \cdot |S(D_i)| < r^0 \cdot (|R(D_i)| - |R'(D_i)|)$ ，则采用 SSJ 策略；否则，采用 DSJ 策略。

如图 1 (b)，对于  $D_2$ 、 $D_3$  区域， $|R_2| > |S_2|$ ， $|R_3| > |S_3|$ ，应从  $S_{site}$  发起空间半连接查询； $D_4$  根据评估结果决定执行策略。

递归区域划分终止的条件是：对于两个参与连接查询的数据集，如果任一数据集在当前区域内的空间对象数目少于某一阈值  $M$ ，则该区域不再作进一步的划分。

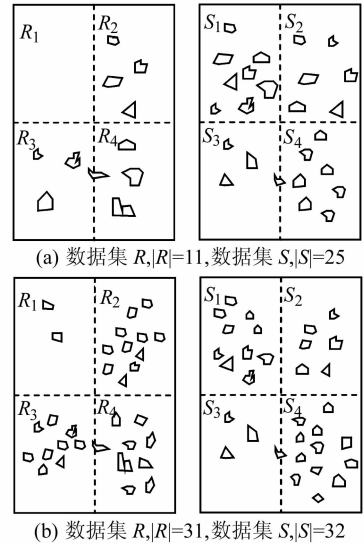


图 1 两个分布的连接数据集  $R$  和  $S$   
Fig. 1 Two Datasets  $R$  and  $S$  at Different Sites

## 2 WFS 服务连接查询优化策略

一般分布式环境和面向 WFS 服务环境下的查询优化问题存在较大差异，文献[3,4]中提出的查询策略不能直接应用于面向 WFS 服务的连接查询。在空间半连接查询和基于区域递归划分的连接查询思路的启发下，本文根据 WFS 服务查询的特点，提出其查询优化的执行策略。

### 2.1 WFS 查询接口

据 OGC 的 WFS 服务实现规范<sup>[5]</sup>，数据的查询是通过 GetFeature 操作来完成的。GetFeature 请求的 ResultsType 属性决定了结果返回类型，当指定为 results 时，服务器将返回满足查询

要求的空间对象,maxFeatures 属性指定返回的空间对象的最大数目.resultsType 属性为 hits 时,返回符合查询条件的空间对象数目  $N$ 。

## 2.2 WFS 连接查询问题分析

WFS 连接查询与文献[1-3]中讨论的问题有较大差异:① 数据集所在的两个站点  $R_{\text{site}}$ 、 $S_{\text{site}}$  不能直接通信,WFS 服务  $S_1$ 、 $S_2$  被动地响应客户的请求;② 用户的连接查询必须通过服务集成站点 ( $I_{\text{site}}$ ) 来协调完成,所有的查询指令都由  $I_{\text{site}}$  发出;③ 需要计算数据分布所耗费的费用;④ 服务器需要把查询结果转化为 GML 格式,而集成端 ( $I_{\text{site}}$ ) 需要把 GML 文档还原为空间对象,耗费的时间量与网络传输数据量可近似看作呈线性关系。

例如,对前面提到的查询问题 1——“找出某给定区域内所有距离消防站不超过 2 km 的商场”。设  $R_{\text{site}}$  提供商场信息  $R$ , $S_{\text{site}}$  提供消防站信息  $S$ ,服务集成站点  $I_{\text{site}}$ ,求取查询结果  $\psi$ 。如果不考虑执行区域划分,有以下三种可行的策略:① 下载  $R$ 、 $S$  到  $I_{\text{site}}$ ,在  $I_{\text{site}}$  求取连接查询结果。② 下载  $S$  到  $I_{\text{site}}$ ,把包含  $S$  的空间属性的查询条件发送到  $R_{\text{site}}$ ; $R_{\text{site}}$  执行查询后把结果发送到  $I_{\text{site}}$ ,查询结果即为满足条件的商场。③ 下载  $S$  到  $I_{\text{site}}$ ,把包含  $S$  数据集中的空间对象的 MBR(即  $B_f$ ) 的查询条件发送到  $R_{\text{site}}$ ; $R_{\text{site}}$  执行查询后把  $R'$  发送到  $I_{\text{site}}$ ,在  $I_{\text{site}}$  求取连接查询结果。

三种策略的网络传输费用分别为:

$$C_1 = s^0 \cdot |S| + r^0 \cdot |R| \quad (5)$$

$$C_2 = 2 \cdot s^0 \cdot |S| + r^0 \cdot |\psi| \quad (6)$$

$$C_3 = s^0 \cdot |S| + mbr^0 \cdot |S| + r^0 \cdot |R'| \quad (7)$$

其实,由于 WFS 服务的封装性,除非  $R$ 、 $S$  是点状地物要素集, $r^0$ 、 $s^0$  无法事先精确获知,但可以通过抽样调查的方式获取。策略②中的  $|\psi|$ 、策略③中的  $|R'|$  按式(3)估算。策略②、③各有一种对称策略,即查询是从  $R_{\text{site}}$  发起的,其网络传输费用分别为:

$$C'_2 = 2 \cdot r^0 \cdot |R| + s^0 \cdot |\psi| \quad (8)$$

$$C'_3 = r^0 \cdot |R| + mbr^0 \cdot |R| + s^0 \cdot |S'| \quad (9)$$

进行优化的目的就是希望从以上五种策略中选择一种费用最低的策略来执行。但是考虑式(6)、式(7),当数据集  $S$  下载到  $I_{\text{site}}$  之前,  $|\psi|$ 、 $|R'|$  根本无法估算。同理,式(8)、式(9)中的  $|\psi|$ 、 $|S'|$  也无法估算,所以只能选择从两个数据集中把数据量较小的先下载  $I_{\text{site}}$ 。对于策略② ( $C_2$ ,  $C'_2$ ),把整个数据集再传到另一个服务,一般情形下不可取。故可以根据式(2)来决定执行下

面两种方式之一:① 直接从另一端下载整个数据集。② 把已下载到  $I_{\text{site}}$  的空间对象的最小包围矩形发送到另一端,执行半连接查询,并取回其结果。

## 2.3 基于递归区域划分的 WFS 连接查询

综合考虑基于递归区域划分的连接查询策略的优点及 WFS 服务查询的特点,本文提出的优化策略如下。

1) 调用 GetFeature(设置 resultsType 属性为 hits)操作,获取整个查询窗口内两个数据集的空间对象数目。同时根据两个数据集在查询窗口中的空间对象集合的最小包围矩形,求取它们的交叉区域(为一矩形区域),把该区域作为划分的起始区域。

2) 判断是否需要进一步划分,如果需要,把查询区域按  $2 \times 2$  网格划分为 4 个区域。对每个子区域编号,如图 1 所示,对其中的三个网格区域执行策略①,最后一个网格区域的对象数目由上一级区域对象数目减去其他三个区域对象数目之和而得。可推知,如果把一个查询区域按上述方法划分为  $N$  个子区域,则对每个服务需要调用  $N$  次 GetFeature 操作计算对象数目。图 1(b)分为 4 个区域,需要调用 4 次,如果把  $R_1$ 、 $S_1$  再作划分,则分为 7 个区域,每个服务调用 7 次。为了减少查询次数,对于每一个子区域,当获取其内部的空间对象数目之后,判断是否有必要继续对下一区域进行探测。判断准则是父区域内的空间对象数目与已探测区域的空间对象数目之差小于  $M$ 。如父区域  $|R| = 50$ ,  $M = 20$ ,若有  $|R_1| = 35$ ,说明  $R_1$  区域的密度较大,而其他剩余区域的空间对象数目已小于  $M$ ,再划分区域探测已经没有必要了,则把剩余的区域合并。需要注意的是,如果把整个查询区域划分为  $N$  个,总共需要探测的次数仍然是  $N$  次,调用服务  $2N$  次。合并策略使得  $N$  的值在网络数据传输费用不增加的情况下尽量减少。

3) 对每个子区域  $D_i$ ,根据  $r^0$ 、 $s^0$ 、 $|R(D_i)|$ 、 $|S(D_i)|$ ,选择把  $R_{\text{site}}$  还是  $S_{\text{site}}$  在  $D_i$  区域的空间对象数据下载到  $I_{\text{site}}$ 。决策准则是选择从  $r^0 \cdot |R(D_i)|$ 、 $s^0 \cdot |S(D_i)|$  中较小者对应的站点开始执行查询,若有  $|R(D_i)|$  或者  $|S(D_i)|$  之一者为 0,则把  $D_i$  从  $\pi(D)$  中剔除。

4) 为了减少服务调用次数,把首先需要从  $R_{\text{site}}$  下载数据的区域划分合并成为  $D^1$ ,相应地,把需从  $S_{\text{site}}$  下载数据的区域划分合并为  $D^2$ 。把  $D^1$ 、 $D^2$  分别作为参数发送到  $R_{\text{site}}$ 、 $S_{\text{site}}$ ,完成第一阶段

的数据下载。

5) 第一阶段的数据下载完成后,在  $I_{\text{site}}$  有了以下数据:①  $R$  数据集中在  $D^1$  区域的空间对象集合  $R(D^1)$ ;②  $S$  数据集中在  $D^2$  区域中的空间对象集合  $S(D^2)$ 。对于  $D^1$  来说,第二阶段的查询任务是找出  $S$  数据集中与  $R(D^1)$  中空间对象满足空间关系  $r.A\theta s.B$  (或者  $f_R(r.A)\varphi f_S(s.B)$ ) 的空间对象集合  $S(D^1)$ ;对于  $D^2$  来说,求  $R(D^2)$ 。

6) 有了各区域内的空间对象,就可以估算  $|R'(D_i)|$  或  $|S'(D_i)|$  了。对于每个区域  $D_i$  而言,确定是直接从另一服务下载数据,还是把已下载的空间对象的最小包围矩形送到另一服务,执行半连接查询。把执行相同策略的区域合并起来,对于  $D_i$  而言,又重新划分成为两个子区域  $D^{11}$ 、 $D^{12}$ 。对  $D^{11}$  区域,直接从  $S_{\text{site}}$  下载该区域对应的空间对象,而对于  $D^{12}$ ,则执行半连接查询策略,即把  $D^{12}$  区域内  $R$  数据集的空间对象的 MBR 提交给  $S_{\text{site}}$ ,得到半连接查询结果,然后在  $I_{\text{site}}$  完成最后的连接。对于  $D^2$  区域,执行同样的策略,将之划分为  $D^{21}$ 、 $D^{22}$  子区域。

### 3 实验结果及分析

本文对基于递归区域划分的 WFS 服务空间连接查询 (PBSM) 策略进行了性能测试,并与 DSJ 策略和 SSJ 查询策略进行了比较分析。

在实验中,设置 3 个计算机节点 A、B、C,分别代表  $R_{\text{site}}$ 、 $S_{\text{site}}$ 、 $I_{\text{site}}$ ,三台机器配置如下:操作系统为 Windows XP Professional sp2, CPU 为 Intel(R) Core(T) 2 Duo CPU T5450,1.66 GHz, 1.00 GB RAM,100 M 网卡。实验环境为 100 M 局域网,每两台机器之间的网络响应速度小于 1 ms。 $M$  取值为 30。实验数据为 Shape File 格式,并建立空间索引。如图 2 所示, $R$ 、 $S$  数据集分别为某市一  $5\text{ km}\times 4\text{ km}$  区域内的房屋与水域分布数据, $R$ 、 $S$  中的空间对象均为面对象, $r^0=7$ , $s^0=13$ ,中间窗口对中间一部分空间对象进行放大显示。实验任务为求取附近 20 m 以内有房屋的水域对象,有 114 个水域对象满足要求。

实验方案如下:

1) 在 A、B 两台计算机安装 ArcEngine9.2, Visual studio. Net 2005, IIS5.1, 利用 C# 语言编写处理查询请求处理程序,部署在 IIS5.1 上。ArcEngine9.2 用于实现对空间数据的查询功能。

2) 计算机 C 安装 ArcEngine9.2, Visual studio. Net 2005, 利用 C# 语言编写执行三种连接查

询策略的客户端程序。

查询窗口范围选择为:① 整个区域,  $5\text{ km}\times 4\text{ km}$ ,  $|R|=654$ ,  $|S|=306$ ;② 左下角区域,  $2.5\text{ km}\times 2\text{ km}$ ,  $|R|=214$ ,  $|S|=102$ ;③ 中部区域,  $2\text{ km}\times 1.5\text{ km}$ ,  $|R|=62$ ,  $|S|=40$ 。表 1 为三种策略在不同范围内的时间费用。实验结果表明,当查询区域范围内包含的数据量越大时,对查询性能的优化效果越明显;当数据量较小时,算法对整个查询区域选择 DSJ 或者 SSJ 策略中的一种,故 PBSM 策略的查询性能接近于两种策略中性能较好的一种。

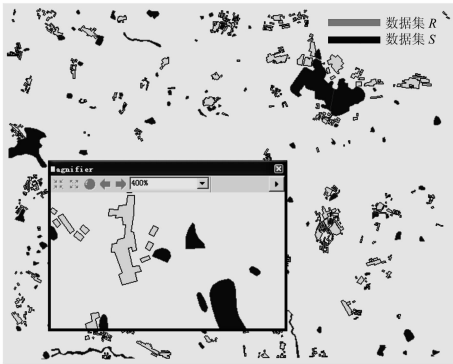


图 2 实验数据集 R 和 S  
Fig.2 R and S for Experiment

表 1 三种策略时间代价/s

Tab.1 Time Costs of Three Strategies/s

查询范围	DSJ	SSJ	PBSM
5 km×4 km	110~120	84~89	72~74
2.5 km×2 km	38~40	27~29	22~24
2 km×1.5 km	14~15	7~8	8~9

整个查询所造成的时间费用由运算费用和数据传输费用两部分构成。运算费用由以下部分组成:① 服务器从空间数据库(或文件)中查询满足要求的空间对象;② 空间对象在服务器端用 GML 编码,在  $I_{\text{site}}$  端解码;③ 在  $I_{\text{site}}$  进行空间连接查询。第一部分取决于空间数据模型、空间索引方式以及服务器的性能,三种策略在该部分所耗费的时间费用差异不大。由于 SSJ、PBSM 策略的过滤作用,对于第三部分,一般情形下, PBSM 策略为最优策略,SSJ 策略次之。当在高速网络环境下,数据传输所需的时间较少,因此运算费用起着主导作用。而当在低速网络环境下,数据传输的时间费用起着主导作用,SSJ、PBSM 策略的过滤率越高,数据传输所需的时间减少得越明显。

### 参 考 文 献

[1] Abel D J, Ooi B C, Tan K L, et al. Spatial Join

Strategies in Distributed Spatial DBMS[J]. Lecture Notes in Computer Science, 1995,951:348-367

[2] Tan K L, Ooi B C, Abel D J. Exploiting Spatial Indexes for Semijoin-Based Join Processing in Distributed Spatial Databases [J]. IEEE Transaction on Knowledge and Data Engineering, 2000, 12 (6): 920-937

[3] 陈莹. 分布式地理空间数据服务集成技术研究[D]. 长沙:国防科学技术大学,2005

[4] Patel J M, De Witt D J. Partition Based Spatial-merge Join[C]. The ACM SIGMOD’96, Montreal, Quebec, 1996

[5] Vretanos P A. Web Feature Service Implementation Specification (OGC 04-094)[OL]. Ver. 1. 1. 0. <http://portal.opengeospatial.org/files/?artifact.id=8339>,2005

第一作者简介:蓝贵文, 博士生,现主要从事 GIS 研究。  
E-mail:gwlan\_770728@163.com

A Spatial Join Strategy Between OGC-Compliant Web Feature Services

LAN Guiwen<sup>1</sup> HUANG Quanyi<sup>2</sup> ZHOU Xiaoping<sup>3</sup>

(1 School of Geodesy and Geomatics, Wuhan University, 129 Luoyu Road, Wuhan 430079, China)

(2 Center for Public Safety Research, Tsinghua University, Haidian District, Beijing 100084, China)

(3 Key Laboratory of Geo-Informatics of State Bureau of Surveying and Mapping, Chinese Academy of Surveying & Mapping, 16 Beitaping Road, Beijing 100039, China)

**Abstract:** We examine the problem of spatial join strategy between OGC-compliant Web feature services (WFS). Based on the semijoin-based join strategy and partition based spatial-merge join strategy, we present a spatial join processing strategy between two OGC-compliant WFS. The experimental result shows that when the cardinality of spatial entities in the query window is large enough, the proposed strategy can reduce the expensive of transmission cost and thus time cost.

**Key words:** Web feature services; spatial join; query optimization; iterative partition

About the first author: LAN Guiwen, Ph.D candidate, majors in GIS.  
E-mail: gwlan\_770728@163.com

(上接第 645 页)

services to improve the recall and the precision of geographical information services, study the multilevel matching arithmetic of geographical information services. This arithmetic filters and matches the services in service register center according as the similarity between services selected with services requested through defining the function similarity and credit standing similarity. The matching arithmetic, geographical information services ontology, and semantic describing form the discovering model. The results show that the ontology-driven discovering model is excellent in recall and precision, and can maintain dynamic loading balance of service copy.

**Key words:** ontology-driven;service discovering;function similarity ;credit similarity;recall; precision;loading balance

About the first author: ZHANG Lichao, Ph.D candidate,majors in GIS and geographical information services.  
E-mail: zhanglichao\_2004@yahoo.com.cn