

2000, 2 (12)
1-6

GIS 地理信息系统 局域网
计算机科学 2000 Vol. 27 No. 12
数据共享

GIS 网络语义共享问题

The Problem of GIS Network Semantic Sharing

黄裕霞¹ 柯正谊² 何建邦³

P208

(中国科学院遥感应用研究所 北京 100101)¹ (浙江大学东南土地管理学院 杭州 310029)²

(中国科学院地理研究所 北京 100101)³

Abstract In this paper, the problem of how to achieve GIS semantic sharing through network has been given. We think this problem is very important. Furthermore, the general way to resolve it has been given.

Keywords GIS network, Semantic sharing, Computer resolving

我们想到了这个问题,先谈谈对它的看法。

1. 问题的提出,问题的说明,这个问题重要不重要?

1.1 问题的提出

①假如说,总理的机要秘书从国务院局域网下载了两个数据,是总理要她了解的。一个是某部门提供的当年全国耕地面积,18.7亿亩;另一个是另一部门提供的,也是当年的全国耕地面积,20.1亿亩。顺便说一句,这两个数据是假设的,像全国耕地面积这样一种极其敏感的数据,机要秘书绝对不会泄露,我们没有必要知道,也不太可能知道。面对这两个耕地面积,总理陷入了沉思,换句话说,有了问题。

②再假如说,甲师傅手头剩有100元现金。一位朋友把公众网上发布的菜市行情告诉了他:西郊农贸市场的白菜批发价格每百斤50元;东郊农贸市场的白菜批发价格每百斤60元。面对这两个白菜价格,甲师傅要盘算盘算,换句话说,也有了问题。

不用说,像这类问题,在网络中只要肯找,准会找出一大批。

1.2 对问题的说明

现在对上面的问题,分析、概括一下:

第一、这些都是GIS网络共享数据。首先,耕地面积是总理和他的机要秘书在国务院局域网上共享的,白菜价格是甲师傅和他的朋友在某市公众网上共享的,甲师傅自然是间接共享。其次,这两个耕地面积是位于北半球的中华人民共和国的耕地面积,这两个白菜价格是西郊和东郊农贸市场的白菜批发价格。所以说,这些都是GIS网络共享数据。

第二、GIS网络数据共享时怎么会出语义共享

问题? 总理想:“一个18.7亿,一个20.1亿,我相信哪个好? 都是我从财政拨出数以千万乃至数以亿计的巨款,经采集、汇总得出的,都是经过专家学者论证可行,并通过鉴定的。看样子我只好两个都信。”

照理说,这两个数据就按为它们花去的钱,也应该都有参考价值。问题是它们各自的参考价值有多大? 这要看它们携带的信息是不是齐备,是不是准确。所以,总理如要知道全国的耕地面积,光知道这“18.7亿亩”和“20.1亿亩”的“全国耕地面积”是远远不够的,还要知道:是不是一个包括了田埂、田间小路和小沟面积,另一个不包括? 这两个数据从采集,到汇总,到上报,各自都花了多长时间? 在这几年中,估计全国耕地面积发生了哪些变化? 还有,这个耕地面积是用大比例尺航片实地调绘、转绘、量算、汇总得来的? 还是用卫片解释抽样调查统计得来的? 还是,……等等。

现在来看甲师傅,光有这两个白菜价格他也不好盘算,还要知道:农贸市场距他住家有多远? 距离允许他摆摊卖菜的地点有多远? 哪个农贸市场的白菜水份少? 口感佳? 卖得出? 等等。

近年来,一种叫作元数据的数据变得日益热门起来。按照FGDC比较权威的定义讲,“元数据是描述数据的数据”,所以它要跟着数据转。您把耕地面积、白菜价格当数据,那么,上面所有那些描述它们或与它们有关的数据,都是它们的元数据。数据本身连同这个数据的所有元数据的含义的集合,我们管它叫这个数据的广义语义。这个数据的定义则叫作它的狭义语义,例如,全国耕地面积的狭义语义是:国家领土范围内种植农作物的全部土地面积,包括或不包括田埂面积,等等。

总理面对两个全国耕地面积陷入沉思的,我们甲

黄裕霞 博士后。

师傅面对两个白菜价格反复盘算的,其实是性能相同的一类问题,即怎样分析和处理 GIS 网络上内容重叠而又不同的数据含义,我们管它叫 GIS 网络的语义共享问题。

对语义分析结果的应用处理不属于这个问题的讨论范围。

第三、这类问题称得上 GIS 网络语义共享问题吗?可能有某位 GIS 专家说:要能从 ARC/INFO 读出全国耕地面积数值,把它放进 MGE 的模型中处理,或者倒过来,从 MGE 出,ARC/INFO 进,总之,我们研究探讨的是分布式异构、异类 GIS 数据库间的数据共享,这才是 GIS 网络共享问题,等等。

可能还会有某位计算机专家说:你们知道什么叫语义吗?语义是,……等等。

我们很赞同这位 GIS 专家和这位计算机专家的看法。没有 GIS 专家的 GIS 网络共享,数据出不来,进不去;没有计算机专家的语义,软件编不成,或编出来的有毛病,这样一来,还谈什么 GIS 语义网络共享?只是我们觉得,GIS 专家的 GIS 网络共享是技术意义上的共享,我们的 GIS 网络语义共享是应用意义上的共享;计算机专家的语义是形式上的语义,我们的语义是内容上的语义。

1.3 这个问题重要不重要?

总理要从这两个耕地面积的广义语义共享中,尽可能估测全国耕地面积到底大概有多少,以便制订尽可能合乎国情的可持续发展战略方针;甲师傅要从这两个白菜价格的广义语义共享中,尽可能选一个农贸市场,以便尽可能多赚几个钱。中国的可持续发展战略方针和中国师傅的生计对我们每个中国人民都很重要,所以,GIS 语义网络共享这个问题很重要。

这个问题既然这么重要,有没有人在考虑它?有。我们在互联网的 10000 多篇文献中找到一位埃及人 Yaser Bishr,写过一篇题名“GIS 互操作中的语义研究”的博士论文,是谈 GIS 数据语义共享的。还有几篇谈数据语义共享,但不是谈 GIS 数据语义共享。美国副总统戈尔在他那篇题名“数字地球”的演讲中,提到与 GIS 语义共享有关的 GIS 互操作和元数据问题。美国的 OGC 组织也已把与这个问题有关的“GIS 语义互操作”列作 2000 年的研究任务。

要解决它还有不少疑问,对这些疑问应有初步认识。

2. 这个问题能不能用计算机解决?用计算机能不能全部解决?以及,有没有必要用计算机解决?

2.1 能不能用计算机解决?

如果能够用计算机解决,则首先要能够形式化,即能够把对这个问题的汉语描述转化成约定的符号公

式;其次是形式化后要能够计算,或叫具备可计算性;再就是计算起来要不花太多时间,或叫计算的复杂度低。此外,还要知道用计算机解决这个问题的条件。

2.1.1 形式化 这个问题能够形式化,可以分两个层次进行:问题整体的形式化和问题环节的形式化。

整体形式化有两个目的,一个是对问题作严格界定,分清什么是 GIS 网络数据语义共享问题,什么不是,无论汉语、英语,即使据说是最最严谨的法语,都不如用公式分辨得清楚。分清问题界限的好处是交流时可以少出现问不对题的场面。另一个目的是为环节形式化作好准备工作。问题的整体形式化实现如下:

①规定集合、变量、算符等要素的记号。

(1) g_i , 变量记号。其中, g_i 表示用户关心的 GIS 网络数据广义语义, i 是 g_i 的编号, i 用自然数从 1 编到 n , 在实际情况中 n 是不大的正整数。

(2) G , 集合记号。 G 的元素是 $g_i, i=1, 2, \dots, n$ 。

(3)算符等记号: R , 相关; $<>$, 不相等; \cap , “交”; Φ , “空”。

②定义:集合 G 是 GIS 网络数据语义共享问题, 当且仅当 G 中至少存在两个元素 g_i 和 g_k , 全部满足下列三个条件: (1) $g_i R g_k$; (2) $g_i <> g_k$; (3) $g_i \cap g_k <> \Phi$ 。

例 1 集合 G 中存在元素 g_i 和 g_k 。 g_i : 网络上有关 A 市地震震级数据的广义语义; g_k : 网络上有关 S 市证券交易所 P 股票收盘价数据的广义语义。假定 A 市地震会影响 S 市证券交易所的 P 股票行情, 因而

$g_i R g_k$ 且 $g_i <> g_k$,

但不满足 $g_i \cap g_k <> \Phi$ 条件。

G 不是 GIS 网络数据语义共享问题。

例 2 集合 G 中存在元素 g_i 和 g_k 。 g_i : A 网上有关 C 市 V 项经济指标增长数据的广义语义; g_k : B 网上有关 C 市 V 项经济指标增长数据的广义语义。 g_i 和 g_k 就像两份省城大报刊登的同一条新华社消息, 一个标点不差, 假定排版师傅极其认真负责, 因而

$g_i R g_k$ 且 $g_i \cap g_k <> \Phi$,

但不满足 $g_i <> g_k$ 条件。

G 不是 GIS 网络数据语义共享问题。

例 3 集合 G 中存在元素 g_i 和 g_k 。 g_i : 公众网上 A 市东郊农贸市场白菜批发价格数据的广义语义; g_k : 公众网上 A 市西郊农贸市场白菜批发价格数据的广义语义。 g_i 和 g_k 完全满足:

$g_i R g_k$; $g_i <> g_k$; $g_i \cap g_k <> \Phi$

三个条件, G 是 GIS 网络数据语义共享问题。

上面定义中的算式条件仅用来表示问题的整体形式化, 对 G 的具体判断要计算 g_i 和 g_k 的嵌套环节变量才能得出, 这将在环节形式化后作出说明。

现在讨论环节形式化问题。

假定:G 是 GIS 网络数据语义共享问题,以下简称 GW。

求解:GW 的环节形式化。

步骤:

①显然,GW 的环节形式化就是对 $g_i(i=j,k)$ 的环节形式化。

②增加下列算符记号:

- (1)=,相等;
- (2)-,算术减,仅用来运算数值变量;
- (3)+,算术加,仅用来运算数值变量;
- (3)MAX(·),表示对括弧内的元素求最大值;
- (4)>,表示空间范围上大于,时间上晚于,质量上高于,数量上多于;
- (5)<,表示空间范围上小于,时间上早于,质量上低于,数量上少于;

- (6)U,“并”;
- (7)C,“包含在”;
- (8)D,“包含”;

③元组和它的环节变量记号:

设 g_i 是元组,记作 $g_i(D_i, S_i, T_i, Q_i)$, $i=j,k$,元组可以嵌套。其中: D_i 是 g_i 元组的第一个环节变量,表示 g_i 中的狭义语义,也就是数据的定义。 D_i 是元组,它是由谓词组成的合式公式,即 $D_i(P_{i1}, P_{i2}, \dots, P_{in})$,其中 P 是从已定义好的谓词库中选择的与 g_i 中狭义语义有关的谓词。谓词中的变量也是已定义好的关键词,关键词包括内涵、外延名词、形容词、动词、量词和否定词等,其中一个数量数据的单位, P 的下标表示它在 D_i 中的排列顺序。例如,对空间数据,它可能包括 $at(a, b)$ (a 在 b 里面), $on(a, b)$ (a 在 b 上面), $between(a, b)$ (a 和 b 之间)等;对属性数据,它可能包括 $include(a, b)$ (包括 a 和 b), $exclude(a)$ (除 a 以外)等。

S_i 是 g_i 元组的第二个环节变量,表示 g_i 中的空间语义,包括范围和位置。 S_i 也是元组,即 $S_i(s_{i1}, s_{i2}, s_{i3}, s_{i4}, s_{i5}, s_{i6})$ 。 s_{i1} 表示体积, s_{i2} 表示面积, s_{i3} 表示长度或到定点的距离, s_{i4} 表示点或形心的 X 坐标, s_{i5} 表示点或形心的 Y 坐标, s_{i6} 表示范围,如行政区范围、经纬网范围等。

T_i 是 g_i 元组的第三个环节变量,表示 g_i 中的时间语义。 T_i 也是元组,即 $T_i(t_{i1}, t_{i2})$; t_{i1} 表示起始时点, t_{i2} 表示终止时点。

Q_i 是 g_i 元组的第四个环节变量,表示 g_i 中的质量语义。 Q_i 也是元组,即 $Q_i(q_{i1}, q_{i2}, q_{i3}, q_{i4}, q_{i5})$ 。 q_{i1} 表示范围精度, q_{i2} 表示位置精度, q_{i3} 表示起始时点精度; q_{i4} 表示终止时点精度, q_{i5} 表示质量水平。

上面所有环节变量的变量都允许是元组。

④元组环节变量(包括嵌套元组的变量)的排序:

S_i, T_i, Q_i 的变量按上面说的次序排列,如果缺少某个变量,要在相应的位置补上“#”号。

⑤元组环节变量和嵌套元组变量的运算:

- (1) g_i 和 g_k 有“ \cap ”、“ \cup ”、“ \subset ”、“ \supset ”运算;
- (2) D_i 和 D_k 有“ \cap ”、“ \cup ”、“ \subset ”、“ \supset ”运算;
- (3) S_i 和 S_k 有“ \cap ”、“ \cup ”、“ \subset ”、“ \supset ”、“-”、“+”、“MAX(·)”、“>”、“<”运算;
- (4) T_i 和 T_k 有“ \cap ”、“ \cup ”、“ \subset ”、“ \supset ”、“-”、“+”、“MAX(·)”、“>”、“<”运算;
- (5) Q_i 和 Q_k 有“ \cap ”、“ \cup ”、“ \subset ”、“ \supset ”、“-”、“+”、“MAX(·)”、“>”、“<”运算。

下面讨论怎样解释整体形式化的条件算式。

(1) $g_i \subset g_k$ 成立,当对 g_i 和 g_k 的嵌套环节变量作“ \cap ”运算时,至少有一对变量的“交”是“空”;

(2) $g_i \cap g_k \subset \Phi$ 成立,当对 g_i 和 g_k 的嵌套环节变量作“ \cap ”运算时,至少 D_i 中关键内涵名词的“交”不是“空”;

要判断 G 是不是 GW,上面两个条件就够,满足 $g_i \cap g_k \subset \Phi$,也就满足了 $g_i R g_k$,但不可以倒过来说:满足 $g_i R g_k$,就满足 $g_i \cap g_k \subset \Phi$ 。

在整体形式化中列入 $g_i R g_k$ 条件的用意,是为了把那些虽然相关但不属于 GW 的问题排除在外。我们的目的是判别 GW,没有必要知道怎样具体检验 $g_i R g_k$ 条件。

2.1.2 可计算性和计算复杂度

1)可计算性。从对 GW 的环节形式化中知道:

① g_i 的各层嵌套变量的个数是有限的;

②其中, S_i, T_i, Q_i 的变量个数基本上是恒定的,不随数据本身变化。 S_i 与 S_k, T_i 与 T_k 和 Q_i 与 Q_k 之间的变量运算大多是一对一的;

③唯独 D_i 的变量个数随数据本身变化,而且,在 D_i 与 D_k 之间,变量的各项运算一般是多对多的,其中最复杂的是“ \cap ”运算。

④对 S_i, T_i, Q_i 的各项运算和对 D_i 的“ \cup ”、“ \subset ”、“ \supset ”运算都比较简单,明摆着能用计算机解决:

⑤对 D_i 与 D_k 的“ \cap ”运算,也就是对 D_i 与 D_k 间各个变量的匹配运算,显然可以转化成图灵机组实现:

不违背一般性,假定 $MAX(jn, kn) = jn$,取 P_j ,用它作第一台图灵机的读写头,把 D_k 的全部变量排成图灵机的存储带。就用这个读写头,沿存储带挨个变量读写匹配,完事后再用 P_n 作第二台图灵机的读写头,仍用原先那条存储带。依此类推,共用了 jn 台图灵机, jn 个读写头,但存储带公用一条。

目前用的计算机都是在图灵机的基础上设计的,凡可以转化成图灵机的问题,都能用计算机解决。其实

这个运算不过是个简单的两层嵌套循环,也不会出现死循环。

⑥可见,GW 具备可计算性。

2)计算复杂度

①对 S, T, Q 各项运算的计算总量通常是恒定的,叫它作 C;

② D_1 的“U”、“C”、“D”运算的计算总量是变动的,叫它作 M;

③ D_1 与 D_2 的“∩”运算,计算总量是 $j_n \times k_n$,可以近似地看作 $(j_n)^2$;

④GW 的计算复杂度 GFD 是: $GFD = (j_n)^2 + M + C$;

⑤在实际情况下, j_n 和 M 都是不大的正整数, GFD 是二阶多项式,可见 GW 的计算复杂度不是很高。

2.1.3 用计算机解决的条件 用计算机解决 GW 的先决条件是数据“广义语义”的标准化,数据“广义语义”包括数据本身和“元数据”的含义。数据标准化大家已经谈得较多,这里只谈“元数据”的标准化。

目前,不同人对元数据有不同理解,有的仅仅把网页上像“A 地有 B 数据免费提供”之类广告叫作元数据,有的把数据库目录或数据字典叫元数据,等等。我们理解的“元数据”前面已经说明,它们是针对眼前具体数据的描述性数据,不管这些“元数据”放在数据库里,元数据库里,数据库目录里,还是数据字典里,只要告诉放在哪里就行。

我们对“元数据”标准化的基本要求是:

①在目前有关部门正在研究制订的元数据标准中,要统一规定元数据的条目和排放次序;

②统一规定哪些“元数据”放在哪里;

③对“元数据”中的狭义语义(数据定义)做文本描述时,要尽可能使用标准化的各类关键词组。要为每个与 GIS 有关的学科或专题选定个数不多的(例如不超过 100 个)关键内涵名词和关键外延名词;关键的形容词、动词、量词、否定词等则各学科专题通用。在数据定义文本中,每个谓词两端要插入统一标准的标记,例如 W;每个关键词两端也都要插入统一标准的标记,例如,关键内涵名词两端插入 N,关键外延名词两端插入 W,关键形容词两端插入 A,关键动词两端插入 V,关键量词两端插入 Q,关键否定词两端插入 F,等等。选定各类关键词是 GIS 标准化部门的工作,插入标记则是建立数据字典或建立元数据库时的工作,这要看把数据定义文本存放在哪里。目前还不可能实现通用的汉语机器识别,规定统一关键词和插入关键词标记后,对进行 GIS 专题的汉语机器识别有帮助。

④有些数据值,对它们归属的数据库、数据文件或

数据字段的元数据不能满足描述要求,需要为它们增加描述信息。换句话说,在特殊场合,元数据要描述到数据值的层次,特别是对非常重要的数据,如全国耕地面积数值等。但一般数据值共用各级数据集的元数据就够。

2.2 用计算机能不能解决 GW 的全部问题?

①凡人解决不了的,计算机肯定解决不了

举例说明什么是 GW 中“人解决不了”的问题。

已知:GW(g_1, g_k)

g_1 : A 部门人口调查数据广义语义(人口,人的个数,6 万,男 50%,女 50%,C 市,1930/10/25,...

g_k : B 部门人口调查数据广义语义(人口,人的个数,6 万,成年 70%,未成年 30%,C 市,1930/10/25,...

任何人不可能从上面的 GW 中准确地知道 C 市 1930 年 10 月 25 日男孩有多少,女孩有多少,计算机也不可能知道。这类问题叫“数据不可分问题”。在 GW 中,肯定还有“人解决不了”的其它类型问题。

②前提下具备的,计算机肯定解决不了

前提下具备,主要是指没有按标准描述 GIS 数据的广义语义,在这种场合,人有可能解决,计算机却不可能解决,或者很难解决。

2.3 有没有必要用计算机解决?

上面说过 GW 很重要,既然重要,就有必要解决,问题是有没有必要用计算机解决?我们认为有必要,理由是:

①专家们大多能一眼看出 GW 的关键在哪里,可惜不是人人都有专家水平,所以要请计算机专家和 GIS 专家一起,把 GW 形式化了,把分析处理 GW 的步骤机械化了,编成了软件。谁按执行命令键,计算机就替谁把 GW 解决了,当然是指好的软件。

②在集成 GIS 网络数据库时,不同专题数据库间往往会出现多义、交叉、重复、含糊的数据,要把它们重组起来做相关分析,这首先要从它们的广义语义中找出相关部分。这是 GW 方面的工作,专家做起来太吃力,不如叫计算机做。

③在 GIS 网络的历史档案数据中,GW 也会经常出现。例如历年来对同一类资源作过多次调查,这项资源在不同时期未必有完全相同的定义,调查技术会有差别,对成果精度的要求也不一样。因此,在对这批历史数据作时序分析时,就有必要通过 GW 操作,把数据广义语义中影响时序分析的因素排除掉。这类工作专家做起来也比较吃力,不如叫计算机做。

④现在有种种数据库叫数据仓库,它的特点是自己不生产数据,专门从各个网络数据库中寻找需要的数

据,拿来现用或存放备用,寻找数据时可以用到 GW 分析方法,例如,若 g_i, g_k 完全相等,那么,拿其中的一个数据就行,不必两个数据都拿来;若两个数据从广义语义看来都派不上用场,就一个不拿,数据仓库要经常地寻找大量数据,哪位专家都不愿意像机器人那样白天黑夜地去找,只好叫计算机去找。

用计算机解决这个问题的思路。

3 这个问题用计算机大体上怎样解决?

3.1 流程图

大体的解决步骤见图 1。

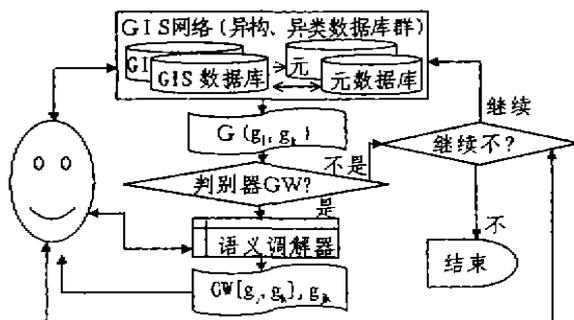


图 1

3.2 流程图说明

3.2.1 用户 图中把用户画成面孔,表示在网络终端上操作的是只想找到 GW 的人。如果用户是数据仓库的“数据提选器”,会把有用的 G 也拿过来,不管它是不是 GW,图中没有画出这种情况。

3.2.2 判别器 语义判别器由“形式化部件”、“判断部件”和“输出部件”组成,用表 1 说明。

表 1

部件	功能	操作
形式化部件	实现 G 的环节形式化。	把从 GIS 网络中找到的 G 的所有变量,按前面说明的标准形式排序。
判断部件	判断 G 是不是 GW?	1)对 g_i 和 g_k 的所有变量一、一对应“交”; 2)用约定标记标识所有取值相等的变量; 3)检验条件: $g_i \langle \rangle g_k$; 4)检验条件: $g_i \cap g_k \langle \rangle \Phi$ 。
输出部件	按不同情况分类输出。	1)G 是 GW,把 GW 传给“语义调解器”; 2)G 不是 GW;不继续,发出“结束”命令; 3)G 不是 GW;继续,发出“再到 GIS 网络找”的命令。

3.2.3 语义调解器 由“预处理部件”、“语义冲突识别部件”、“语义冲突处理部件”、“综合部件”和“输出部件”组成,用表 2(见 P6)说明。

3.2.4 $GW[g_i, g_k]$ 对 $G(g_i, g_k)$ 用表 2 中的“预处理部件”、“语义冲突识别部件”和“语义冲突处理部件”依次处理,得出的结果是 $GW[g_i, g_k]$ 。

3.2.5 g_k 是经过表 2 中“综合部件”处理的结果。

3.2.6 其它补充说明 ①直到现在,为了说明方便,往往把集合 G 看成只包含两个元素, g_i 和 g_k 。实际情况下 G 的元素可能不只两个,这也能够用计算机解决。例如,若元素个数 n 大于 2,可以用 n 台(每台有 n-1 条存储带)图灵机解决。容易证明,这时的计算复杂度还是二阶多项式。②图中当 G 不是 GW,而且不再继续找,就结束了。已经取出的 G 怎么处理?让它去好了。用户只是把它读出来, GIS 网络中并没有少了 G。

我们已经初步设计了怎样用计算机解决这个问题的具体方法,在另一篇文章中谈。

参考文献

- 1 Bishr Y. Semantic Aspects of Interoperable GIS, ITC Publication Series No. 56, ISBN 90 6164 1411, 1997
- 2 阿尔·戈尔. 数字地球——对 21 世纪人类星球的理解. 地理信息, 1998, 7(2)
- 3 Kroenke D M 著, 施伯乐, 等译校. 数据库处理基础、设计与实现. 电子工业出版社, 1998
- 4 OGC. Promoting distributed geoprocessing through cooperative technology development, partnerships and industry consensus. Newsletter of Open-GIS-Consortium, Inc. Wayland, MA, 1996
- 5 Goodchild M F, et al. Interoperating GISs, Report of a Specialist Meeting Held Under the Auspices of the Varenus Project Panel on Computational Implementations of Geographic Concepts. Available at: <http://ncgia.ucsb.edu/conf/interop97>
- 6 Sheth A, Kashyap V. So Far (Schematically) yet So Near (Semantically). In: Proc. of the IFIP TC2/WG2. 6 Conference on Semantics of Interoperable Database Systems, DS-5, Nov. 1992
- 7 Goh C H, et al. Context Interchange: Overcoming the Challenges of Large-Scale Interoperable Database Systems in a Dynamic Environment. In: Proc. of the Third International Conference (CIKM-94), Gaithersburg, Maryland, 1994

表 2

部件	功能	操作
预处理部件	一般性分析以及对分析结果的处理。	<ol style="list-style-type: none"> 1. 对经过“\cap”运算的 S_i, T_i, Q_i 中不相等对应变量取值进行“$>$”、“$<$”、“\supset”、“\subset”和“$=$”等各类运算; 2. 根据运算结果给这些变量标识约定的标记; 3. 在各层元组中添加可能有的“差值”变量值。
语义冲突识别部件 这个部件一般要结合谓词推理、“专家系统”或其它人工智能技术设计,还要有人机对话的能力。	分析识别对应变量值间语义不一致性。	<ol style="list-style-type: none"> 1. 对经过“\cap”运算的 D_i 中不相等对应变量取值进行“语义矛盾”、“语义交叉”、“语义重复”、“语义含糊”等语义不一致性分析识别; 2. 对经过分析识别的变量取值标识约定的标记。
语义冲突处理部件 这个部件一般要结合谓词推理、“专家系统”或其它人工智能技术设计,还要有人机对话的能力。	处理经过语义冲突识别部件分析识别的对应变量值。	<ol style="list-style-type: none"> 1. 对“语义矛盾”的对应变量值进行处理; 2. 对“语义交叉”的对应变量值进行处理; 3. 对“语义重复”的对应变量值进行处理; 4. 对“语义含糊”的对应变量值进行处理;
综合部件 这个部件一般要结合谓词推理、“专家系统”或其它人工智能技术设计,还要有人机对话的能力。	对经过语义冲突处理部件处理的对应变量值进行综合处理。	<ol style="list-style-type: none"> 1. “\cup”运算; 2. “$+$”运算; 3. 增加变量值中应该增加的内容; 4. 删除变量值中应该删除的内容; 5. 标识变量值中矛盾的内容; 6. 其它。
输出部件	分类输出。	<ol style="list-style-type: none"> 1. 输出 $GW[g, g_k]$; 2. g_k。

(上接第 19 页)

参 考 文 献

- 1 Du Jiancheng, Chen Daoxu, Xie Li. JAPS: An Automatic Parallelizing System Based on JAVA. Science in China, 1999, 3: 279~288
- 2 Sarmenta L F G, Hirano S, Ward S A. Towards Bayesian: Building an Extensible Framework for Volunteer Computing Using Java. ACM 1998 Workshop on Java for High-Performance Network Computing, Palo Alto, California, 1998. Concurrency: Practice and Experience, 1998, 10: 1015~1019
- 3 Thornley B J, Ellenbecher M. An Initial Evaluation of the Tera ultithreaded Architecture and Programming System Using the C3I Parallel Benchmark Suite. SC98, Orlando, Florida, November, 1998. 7~13
- 4 Edwards A. The Specification and Execution of Heterogeneous Synchronous Reactive Systems. [Ph. D. thesis]. University of California, Berkeley, May 1997
- 5 Girker M B. Functional Parallelism Theoretical Foundations and Implementations. University of Illinois at Urbana-Champaign. [CSR Report]. 1991. 5~19
- 6 Polychronopoulos C D. Parallel Programming and Compilers. Boston, Kluwer, Academic Pub., 1998. 83~111