



(12)发明专利

(10)授权公告号 CN 103473247 B

(45)授权公告日 2017.02.08

(21)申请号 201210325799.2

(22)申请日 2012.09.05

(65)同一申请的已公布的文献号
申请公布号 CN 103473247 A

(43)申请公布日 2013.12.25

(66)本国优先权数据
201210222442.1 2012.06.30 CN

(73)专利权人 中国地质调查局西安地质调查中心
地址 710054 陕西省西安市友谊东路438号

(72)发明人 王占昌 颜世强 丁克永

(74)专利代理机构 北京市盛峰律师事务所
11337

代理人 赵建刚

(51)Int.Cl.

G06F 17/30(2006.01)

(56)对比文件

US 2010/0094950 A1,2010.04.15,
US 2006/0005233 A1,2006.06.05,
CN 1776622 A,2006.05.24,
CN 101004798 A,2007.07.25,

审查员 韩孟

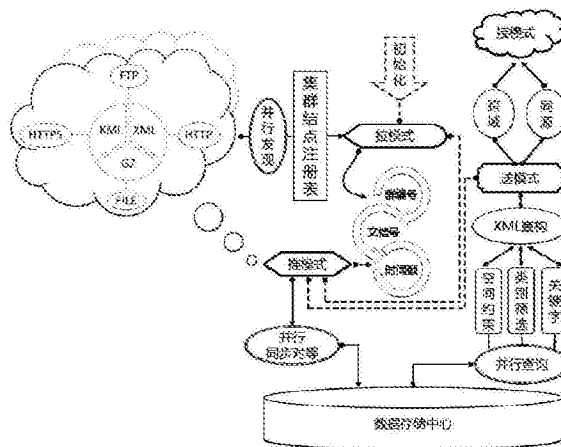
权利要求书1页 说明书15页 附图7页

(54)发明名称

地质资料信息集群机理与接口聚合系统

(57)摘要

一种地质资料信息集群机理与接口聚合系统,包括拉模式、推模式、接模式和送模式四个集群模式;拉模式采集集群节点注册表信息建立三维字典,并建立集群组网形态或集群架构映射树;推模式依据集群架构映射树定位某集群所处的远程路径,进行并行化查询,通过查询结果判断是否能够获取所述远程路径的文档流,如果能够获取所述远程路径的文档流,则将远程路径的文档流保存在数据存储中心;送模式通过接口指令调用请求,对保存在所述数据存储中心的数据执行SQL查询操作,并将查询结果发送到调用接口;接模式在浏览器端向所述远程路径接口服务发送请求。本发明消除了信息孤岛并实现结点群互联互通,有利于避免了地质工作重复投入、减少成本。



1. 一种地质资料信息集群机理与接口聚合系统,其特征在于:包括拉模式、推模式、接模式和送模式四个集群模式;

所述拉模式采集集群节点注册表信息建立三维字典,并建立集群组网形态或集群架构映射树;

所述推模式依据所述集群架构映射树定位某集群所处的远程路径,进行并行化查询,通过查询结果判断是否能够获取所述远程路径的文档流,如果能够获取所述远程路径的文档流,则将所述远程路径的文档流保存在数据存储中心;

所述送模式通过接口指令调用请求,对保存在所述数据存储中心的数据执行SQL查询操作,并将查询结果发送到调用接口;

所述接模式在浏览器端向所述远程路径接口服务发送请求。

2. 根据权利要求1所述的地质资料信息集群机理与接口聚合系统,其特征在于:所述拉模式采取并行技术进行触碰发现和抓取,所述三维字典由集群编号、地质资料XML文档序号和存档时间戳建立三维字典。

3. 根据权利要求2所述的地质资料信息集群机理与接口聚合系统,其特征在于:所述推模式对所述远程路径的定位以所述存档时间戳为纽带。

4. 根据权利要求1所述的地质资料信息集群机理与接口聚合系统,其特征在于:所述送模式依据外界提供的关键字匹配、空间范围约束和类别筛选条件等多种REST接口指令调用请求,通过并发控制和并行查询机制,对集群数据存储中心关系型数据库按海量分页、分类控制和正则匹配算法执行SQL查询操作。

5. 根据权利要求1所述的地质资料信息集群机理与接口聚合系统,其特征在于:所述接模式在浏览器端通过JavaScript脚本向远程接口服务发送请求,浏览器与服务器两者间的通讯采用AJAX技术实现同源访问,采用Web Service代理技术解决因浏览器安全策略对集群结点接口跨域访问的限制,在结点群服务器之间使用HttpWebRequest类进行通讯、通过HttpWebResponse类接收应答。

6. 根据权利要求1所述的地质资料信息集群机理与接口聚合系统,其特征在于:所述集群架构映射树具备动态性,基于集群文档时间戳的变更或失效机制所反映的远程事件和行为主要包括:某群为新增、某群已退出或已被删除、某群编号或次序有变更、某群访问链路中断、某文档被替代、某文档被删除、某文档内容发生变化和某文档读取出现异常。

地质资料信息集群机理与接口聚合系统

技术领域

[0001] 本发明涉及地质调查技术领域,尤其涉及一种具有空间位置的地质资料信息集群“推拉接送”整装接口机理与聚合系统。

背景技术

[0002] 长期以来,地质工作形成的大量地质资料为社会发展和国民经济建设提供了重要支撑,在矿产资源勘查开发和地球科学研究方面所起的导向作用尤为显著,但因分散式信息孤岛问题的存在而导致地质资料信息服务渠道不畅、能力不强、领域不宽等诸多弊端日趋凸显,制约着地质资料信息潜在价值的充分发挥,因此,在统一标准体系下构筑集群共享服务平台,从而实现地质资料信息服务渠道和服务方式多元化十分重要。

[0003] 检索显示,在集群机理方面与本发明接近的同类技术主要涉及服务器集群、数据库集群和高性能计算等领域,如专利号:200510087857.2、200510087855.3、200610170452.X和200610170452.X等,此类专利成果侧重于硬件连接、数据存储分布、编程指令优化、节点状态监控或者负载均衡等基础环境的搭建方面,未涉及地质资料信息(尤其是空间信息)专业应用范畴;检索还表明,基于网络环境信息传递采取的技术措施符合“推拉”模式的应用并不鲜见,已出现在多篇学术论文中,如:基于区域的信息推送服务模式探讨-《情报探索》2011年09期,但集群化“推拉接送”整装接口继承模式属本发明首次提出。

发明内容

[0004] 本发明的目的在于设计一种新型的地质信息集群机理与接口聚合系统,解决上述问题。

[0005] 为了实现上述目的,本发明采用的技术方案如下:

[0006] 一种地质资料信息集群机理与接口聚合系统,包括拉模式、推模式、接模式和送模式四个集群模式;

[0007] 所述拉模式采集集群节点注册表信息建立三维字典,并建立集群组网形态或集群架构映射树;

[0008] 所述推模式依据所述集群架构映射树定位某集群所处的远程路径,进行并行化查询,通过查询结果判断是否能够获取所述远程路径的文档流,如果能够获取所述远程路径的文档流,则将所述远程路径的文档流保存在数据存储中心;

[0009] 所述送模式通过接口指令调用请求,对保存在所述数据存储中心的数据执行SQL查询操作,并将查询结果发送到调用接口;

[0010] 所述接模式在浏览器端向所述远程路径接口服务发送请求。

[0011] 优选的,所述拉模式采取并行技术进行触碰发现和抓取,所述三维字典由集群编号、地质资料XML文档序号和存档时间戳建立三维字典。

[0012] 优选的,所述推模式对所述远程路径的定位以所述存档时间戳为纽带。

[0013] 优选的,所述送模式依据外界提供的关键字匹配、空间范围约束和类别筛选条件

等多种REST接口指令调用请求,通过并发控制和并行查询机制,对集群数据存储中心关系型数据库按海量分页、分类控制和正则匹配算法执行SQL查询操作。

[0014] 优选的,所述接模式在浏览器端通过JavaScript脚本向远程接口服务发送请求,浏览器与服务器两者间的通讯采用AJAX技术实现同源访问,采用Web Service代理技术解决因浏览器安全策略对集群结点接口跨域访问的限制,在结点群服务器之间使用HttpWebRequest类进行通讯、通过HttpWebResponse类接收应答。

[0015] 优选的,所述集群架构映射树具备动态性,基于集群文档时间戳的变更或失效机制所反映的远程事件和行为主要包括:某群为新增、某群已退出或已被删除、某群编号或次序有变更、某群访问链路中断、某文档被替代、某文档被删除、某文档内容发生变化和某文档读取出现异常。

[0016] 本发明的有益效果可以总结如下:

[0017] 1.本发明是实现网络环境下面向公众提供共享服务的核心组件之一,是地学海量空间信息技术瓶颈的有效解决方案,也是合理消除信息孤岛并实现结点群互联互通的关键所在。

[0018] 2.本发明通过模型嵌入或接口调用所获取到的系列化、权威性地质资料信息服务,不仅可提高地质工作社会影响力和公共服务能力,还有利于避免地质工作重复投入、减少社会投资浪费并有助于商业勘查资本的注入,从而提高地质资料信息服务产品的增值。

附图说明

[0019] 图1是本发明地质资料信息集群机理与接口聚合系统的“推拉接送”类体系继承与耦合关系图。

[0020] 图2是本发明地质资料信息集群机理与接口聚合系统在地学空间信息检索系统主流程的嵌入位置图(粗虚线框圈定部分)示意图。

[0021] 图3是本发明地质资料信息集群机理与接口聚合系统拉模式运行流程图。

[0022] 图4是本发明地质资料信息集群机理与接口聚合系统推模式运行流程图。

[0023] 图5是本发明地质资料信息集群机理与接口聚合系统送模式运行流程图。

[0024] 图6是本发明地质资料信息集群机理与接口聚合系统接模式运行流程图。

[0025] 图7是本发明地质资料信息集群机理与接口聚合系统数据存储中心主表、详表级联关系图。

具体实施方式

[0026] 为了使本发明所解决的技术问题、技术方案及有益效果更加清楚明白,以下结合附图及实施例,对本发明进行进一步详细说明。应当理解,此处所描述的具体实施例仅仅用以解释本发明,并不用于限定本发明。

[0027] 实施例一:

[0028] 如图1所示的本发明地质资料信息集群机理与接口聚合系统的“推拉接送”类体系继承与耦合关系图,本发明围绕集群机理与接口聚合系统按“推拉接送”四个模式分述如下:

[0029] 1、拉模式

[0030] 依据集群注册表,以群组为处理单元,针对多协议(如http、https、ftp、file等)采取CPU多核并行技术进行触碰发现和抓取,由集群编号、地质资料XML文档序号和存档时间戳联合组成的哈希键值建立三维字典,形成分布式集群组网形态或架构映射树。其中, file协议强行指向受保护的.\App_Data\xmlpublish路径,以便支持发布池以便支持发布池(自留地)。

[0031] 本发明点在模型系统中处于“基类”位置,其替代技术方案可为“网络爬虫”。

[0032] 2、推模式

[0033] 依据集群架构映射树以集群文档时间戳为纽带快速定位到某群所处的远程路径,进行并行化查询操作,判别是否可按明文或暗文压缩方式获取远程路径内的若干文档流,实现树结构到关系模型(层叠二维表)的转换。

[0034] 为使地质资料文档内“点、线、面、贴图”细粒度空间实体支持访问接口XPath—“XML轴路径查询微语言”快速遍历技术,对每个线图元和面图元进行几何拓扑(长度、周长、面积、范围、内点)计算并将结果存入属性字段表,采取“长事务”机制将空间实体按分类结构推入服务器端SQL数据存储中心,完成正向同步过程。其中,地质体“内点”算法支持“湖中多岛”概念,能返回任意形状各向均质多边形的重心位置,可用于板块动力学和地质灾害滑坡体等方面的量化研究。

[0035] 由于集群架构映射树具备动态性,基于时间戳的变更或失效机制所反映的远程事件和行为包括:某群为新增、某群已退出或已被删除、某群编号或次序有变更、某群访问链路中断、某文档被替代、某文档被删除、某文档内容发生变化、某文档读取出现异常等症状。为解决此类问题采取了反向并行对等技术,包括整理并优化数据库、聚合重构各类关系表、回收无效的存储碎片、重建数据库Btree和GIST索引,以便使分布式集群架构与数据存储中心保持一致并提高检索效率。

[0036] 本发明点在模型系统中处于一级“派生类”位置,其替代技术方案可为“网格计算”或“弹性计算”。

[0037] 3、送模式(基于Microsoft.NetFramework WCF技术的REST接口服务)

[0038] 依据外界提供的关键字匹配、空间范围约束和类别筛选条件等多种REST(表述性状态转移)接口指令调用请求,通过并发控制和并行查询(PLINQ)机制,对集群数据存储中心关系型数据库按海量分页、分类控制和正则匹配算法执行相应的SQL查询操作。

[0039] 其中,关键字列表查询可按“并且”、“或者”两种运算逻辑对图元属性内容进行模糊匹配;空间范围约束查询支持鼠标点击或拉框以及专题图层之间呈叠置、落入、相交或接触关系的GIS空间分析运算功能;类别筛选查询支持数据仓、数据集(如:成矿带、行政区划、盆地流域)、数据库(如:区域地质、矿产勘查等专业类别)、图层(如:比例尺分幅、矿种)级联关系枚举分类检索。

[0040] 查询返回的结果经聚合聚类发送至调用接口,完成由SQL关系数据库向整装(以当前结点为轴沿父代路径递归回溯至根结点)XML结构树的再造与重构。

[0041] 送模式所有功能均以REST指令方式宿主于Windows IIS服务,外界以HTTP路由协议访问,这不仅提供了多元异构地质资料信息跨平台访问、交换和共享解决方案,还通过分页机制有效解决了矢量型地学空间信息“海量”效应(如海量检索、海量传输、海量展现等)引发的诸多技术瓶颈。

[0042] 该类接口的可定制性、可访问性和可伸缩性大大降低了集群“二次开发”技术和地质资料信息“再利用”的难度,为地质资料的跨行业应用提供了底层可嵌入式接口范式和设计准则。与此同时,该类接口针对外界访问模糊了结点群软硬件环境边界,可将用户的关注点直接锁定到地质资料信息丰富的专业内涵和呈现的功能服务方面。

[0043] 本发明点在模型系统中处于二级“派生类”位置,其REST接口替代技术方案可为“SOAP”。

[0044] 4、接模式(接口聚合器)

[0045] 在浏览器端通过JavaScript脚本向远程接口服务发送请求,浏览器与服务器两者间的通讯采用AJAX技术实现同源访问,采用Web Service代理技术解决因浏览器安全策略对集群结点接口跨域访问的限制;在结点群服务器之间使用HttpRequest类进行通讯、通过HttpResponse类接收应答,以便使地质资料信息传输可穿越异构网络环境。

[0046] 通过异步调用接收远程应答的地质资料空间信息,为界面层提供解析、呈现和交互服务。

[0047] 本发明点在模型系统中处于远端界面层位置。

[0048] 实施例二:

[0049] 本发明采用面向用户群体、面向数据资源、面向协同服务的架构思想,在参考OGC国际开放标准的基础上,将具有空间定位特征的多元异构地质资料信息进行统一化描述,使其便于整合、共享和发布;依托高效的并行计算前沿理论,将结点组网模型映射成具松散耦合(弹性)特征的网状或树状“群组”体系,每个独立群或派生群均由“推拉接送”接口聚合器构成,向外界呈现浏览器对结点群(Browser vs Nodes)的访问态势。依时间顺序将工作流程划分为格式制定、模型设计、测试评估和部署运行四个主要阶段。分述如下:

[0050] 1、格式制定

[0051] 为使集群节点间数据便于存储、交换和解析,在参考OGC国际标准基础上,将地质资料信息按“点、线、面、贴图”四类特征进行统一化描述和细粒度划分,形成金字塔状弹性分类结构树:1个数据仓—N个数据集—N个数据库—N个数据层—N个点线面贴图(参数、属性、权限)。部分节点元素定义与OGC推荐的KML、GML保持适度兼容并进行了功能性扩充,尤其体现在安全策略、几何范围、拓扑属性等方面。其中,几何范围<LatLonBox>属于动态元素,目的是为基于“轴路径”的XPath快速遍历提供空间约束参数;<visibility>元素定义为四态键,依次记录并控制分类树在界面层是否可视、半可视、不可视、未发生获取;<open>元素定义为六态键:0=关闭态(服务器端:未读取)、1=开启态(服务器端:未读取)、pages+page+2=关闭态(服务器端:已读取)、pages+page+3=开启态(服务器端:已读取)、4=关闭态(本地:已读取)、5=开启态(本地:已读取),用于控制分类树如何呈现在界面层并记录异步通讯事件。

[0052] 文件类型采用XML扩展名,此类文件已由操作系统在类型关联(MIME)中预定义为“text/xml”,可自由穿越网络防火墙实现跨平台访问。

[0053] 以下是本发明专用的XML架构定义及说明:

[0054] <?xml version="1.0" encoding="utf-8"?>

[0055] <geo>

[0056] <Dataware><!--数据仓定义-->

- [0057] <name>GEOSite</name><!--数据仓名称-->
- [0058] <description>GEOSite工作区</description><!--数据仓描述-->
- [0059] <visibility>1</visibility><!--visibility为0/1/2/3四态,指示是否可视、半可视、不可视、未发生获取,默认值0-->
- [0060] <open>1</open><!--open为0/1,指示是否打开,忽略则取默认值1-->
- [0061] <Dataset><!--数据集定义-->
- [0062] <name>数据集名称</name>
- [0063] <description>数据集描述</description>
- [0064] <visibility>1</visibility><!--visibility为0/1/2/3四态,指示是否可视、半可视、不可视、未发生获取,默认值0-->
- [0065] <open>1</open><!--open为0/1,指示是否打开,忽略则取默认值1-->
- [0066] <Database><!--数据库定义-->
- [0067] <name>数据库名称</name>
- [0068] <description>数据库描述</description>
- [0069] <visibility>1</visibility><!--visibility为0/1/2/3四态,指示是否可视、半可视、不可视、未发生获取,默认值0-->
- [0070] <open>1</open><!--open为0/1,指示是否打开,忽略则取默认值1-->
- [0071] <Layer><!--图层定义-->
- [0072] <name>图层名称</name>
- [0073] <description>图层描述</description>
- [0074] <visibility>1</visibility><!--visibility为0/1二态,指示是否可视、不可视,默认值0-->
- [0075] <open>1</open><!--open为0/1/2/3/4/5六态,0=关闭态(服务器端:未读取)1=开启态(服务器端:未读取)pages+page+2=关闭态(服务器端:已读取)pages+page+3=开启态(服务器端:已读取)4=关闭态(本地:已读取)5=开启态(本地:已读取),默认值0-->
- [0076] <Element id="内部图元序号(0)"category="Point"rank="访问级别号(0~3)默认0rank=0可编辑=1可查看属性2=可浏览提示3=仅显示图形-服务器端"><!--点要素定义,支持自定义子图符号--><!--在点要素中,Point、center和LatLonBox均采用了相同的坐标对,虽出现信息冗余,却为XPATH轴遍历法提供了便利-->
- [0077] <name from="字段序号,字段序号,,,默认0,-1可忽略字段"center="十进制经度,十进制纬度">点图元快捷提示默认为空</name>
- [0078] <description>
- [0079] <field name="字段名称默认【field】"alias="字段别名默认【描述信息】">字段内容默认为空</field>
- [0080] </description>
- [0081] <Point marker="点图元子图编号(0~?)默认0子图均可在服务器端自定义">
- [0082] <coordinates>十进制经度,十进制纬度</coordinates>
- [0083] </Point>
- [0084] <LatLonBox>

[0085] <north>(上)十进制纬度</north>

[0086] <south>(下)十进制纬度</south>

[0087] <west>(左)十进制经度</west>

[0088] <east>(右)十进制经度</east>

[0089] </LatLonBox>

[0090] </Element>

[0091] <Element id="内部图元序号(1)"category="Line"rank="访问级别号(0~3)默认0rank=0可编辑=1可查看属性2=可浏览提示3=仅显示图形-服务器端"><!--线要素定义,支持参数控制和属性结构定义-->

[0092] <!--在线要素中,name节中的center为线元中点坐标对-->

[0093] <name from="字段序号组成的逗号分隔序列,若此序列含负值,则强行取name,并忽略字段;若某序号并不存在,则跳过该值"center="十进制经度,十进制纬度">提示信息默认为空</name>

[0094] <description>

[0095] <field name="字段名称"alias="字段别名">字段内容默认为空</field>

[0096] </description>

[0097] <LineString width="线宽度(1~10)默认1"alpha="线透明度(0~100)默认100"rgb="线颜色红绿蓝默认#ff0000">

[0098] <coordinates>十进制经度,十进制纬度空格...,...,...空格十进制经度,十进制纬度</coordinates>

[0099] </LineString>

[0100] <LatLonBox>

[0101] <north>(上)十进制纬度</north>

[0102] <south>(下)十进制纬度</south>

[0103] <west>(左)十进制经度</west>

[0104] <east>(右)十进制经度</east>

[0105] </LatLonBox>

[0106] </Element>

[0107] <Element id="内部图元序号(2)"category="Polygon"rank="访问级别号(0~3)默认0rank:0=可编辑(移动、删除、修改)1=可查看属性(属性浏览+提示)2=可浏览(提示)3=仅显示图形(提示)-服务器端"><!--面要素定义,参数定义和结构定义--><!--在面要素中,name节中的center为面元最佳内点坐标对-->

[0108] <name from="字段序号组成的逗号分隔序列,若此序列含负值,则强行取name,并忽略字段;若某序号并不存在,则跳过该值"center="十进制经度,十进制纬度">提示信息</name>

[0109] <description>

[0110] <field name="字段名称"alias="字段别名">字段内容默认为空</field>

[0111] </description>

[0112] <Polygon>

- [0113] <outerBoundaryIs>
- [0114] <LinearRing width="边界线宽度(1~10)默认1"alpha="边界线透明度(0~100)默认100"rgb="边界线颜色红绿蓝默认#ff0000"fillalpha="面透明度(0~100)默认20"fillrgb="面颜色默认#0000ff">
- [0115] <coordinates>(起始点)十进制经度,十进制纬度空格...,...,...空格(最末点)十进制经度,十进制纬度(起始点)十进制经度,十进制纬度</coordinates>
- [0116] </LinearRing>
- [0117] </outerBoundaryIs>
- [0118] <innerBoundaryIs><!--“湖中多岛”-->
- [0119] <LinearRing>
- [0120] <coordinates>(起始点)十进制经度,十进制纬度空格...,...,...空格(最末点)十进制经度,十进制纬度(起始点)十进制经度,十进制纬度</coordinates>
- [0121] </LinearRing>
- [0122] <LinearRing>
- [0123] <coordinates>(起始点)十进制经度,十进制纬度空格...,...,...空格(最末点)十进制经度,十进制纬度(起始点)十进制经度,十进制纬度</coordinates>
- [0124] </LinearRing>
- [0125] </innerBoundaryIs>
- [0126] </Polygon>
- [0127] <LatLonBox>
- [0128] <north>(上)十进制纬度</north>
- [0129] <south>(下)十进制纬度</south>
- [0130] <west>(左)十进制经度</west>
- [0131] <east>(右)十进制经度</east>
- [0132] </LatLonBox>
- [0133] </Element>
- [0134] <Element id="内部图元序号(2)"category="Image"rank="访问级别号(0~3)默认0rank:0=可编辑(移动、删除、修改)1=可查看属性(属性浏览+提示)2=可浏览(提示)3=仅显示图形(提示)-服务器端"><!--在贴图要素中,name节中的center为范围中点坐标对-->
- [0135] <name from="字段序号组成的逗号分隔序列,若此序列含负值,则强行取name,并忽略字段;若某序号并不存在,则跳过该值"center="十进制经度,十进制纬度">提示信息</name>
- [0136] <description>
- [0137] <field name="字段名称"alias="字段别名">字段内容默认为空</field>
- [0138] </description>
- [0139] <Icon>
- [0140] <href><http://www.map.com/? .png></href>
- [0141] </Icon>

[0142] <LatLonBox>
[0143] <north>(上)十进制纬度</north>
[0144] <south>(下)十进制纬度</south>
[0145] <west>(左)十进制经度</west>
[0146] <east>(右)十进制经度</east>
[0147] </LatLonBox>
[0148] </Element>
[0149] </Layer>
[0150] </Database>
[0151] </Dataset>
[0152] </Dataware>
[0153] </geo>
[0154] 2、模型设计

[0155] 在Microsoft Visual Studio 2010开发环境内采用C#语言编写,主要支撑技术包括:REST(表述性状态转移)、WCF(视窗通讯基础)、PLINQ(并行语言集成查询)、Parallel(并行化:任务并行、数据并行)。

[0156] 说明书附图中的图1是本发明所涉及“推拉接送”模式各模块之间的继承关系和运行流程图。模型定义的类型体系自基类至派生类依次为:拉模式类、推模式类、接模式类、送模式类。该模型基于WCF技术实现,类体系承载于Windows IIS服务中,初始化过程通过注册路由的方式进行;由集群号、文档号、时间戳组成的三维字典反映了集群节点组网虚拟形态,是拉模式与推模式的连接纽带;服务器端数据存储中心是推模式与接模式的访问枢纽;图2中所示的几个并行体主要针对具有枚举特征的多个群组和多条地质体记录而展开,从而极大提高了多核服务器的运行性能。

[0157] 图2是本发明地质资料信息集群机理与接口聚合系统在地学空间信息检索系统主流程的嵌入位置图(粗虚线框圈定部分)示意图。

[0158] 图3是本发明地质资料信息集群机理与接口聚合系统拉模式运行流程图。其中,集群注册表由组网人员按地质资料涉密审批流程创建本地或远程链接并对可发布文档进行统一组织和命名。经手工或自动调用对等指令,通过发现机制完成三维字典的创建。该字典仅在内存中开辟,不进行物理存储。

[0159] 图4是本发明地质资料信息集群机理与接口聚合系统推模式运行流程图。通过手工或自动调用对等指令,将字典中的时间戳与数据存储中心关系表进行对比,执行正向同步过程并存储空间要素的拓扑信息以便保持两者的一致性;然后将字典中的时间戳与远程结点群组进行对比,执行反向对等并重建索引关系,从而确保集群架构与数据库之间严格匹配。

[0160] 图5是本发明地质资料信息集群机理与接口聚合系统送模式运行流程图。将接口指令传输的访问请求转换为标准SQL语句,通知数据存储中心执行该语句并返回结果集,此结果涉及的海量问题通过在SQL语句中附加分页控制查询参数进行解决。传送出的XML数据经过了类别聚合与节点重构以便形成完整的地质资料分类树供解析使用。

[0161] 图6是本发明地质资料信息集群机理与接口聚合系统接模式运行流程图。远程接

口服务按服务器端后台和浏览器端前台两者调用方式进行。后台调用采用.net提供的类实现通讯,前台调用采用JavaScript语言提供的AJAX技术实现,浏览器端脚本执行面临跨域障碍,本模型采用了服务器端Web Service代理迂回策略解决了跨域问题。通过异步接收,对封装的XML数据依据正则算法进行过滤提取。

[0162] 图7是“地质资料信息集群机理与接口聚合系统”数据存储中心主表、详表级联关系图。矢量型地质资料空间信息的存储策略制约着海量信息访问、检索、运算等功能的实现和执行效率,本模型将关系数据库划分为五个呈现级联关系的二维表:数据仓、数据集、数据库、图层、图元。当首次调用对等指令时,将采取“长事务”机制自动建立各关系表,同时还依据字段的关联性自动建立B树联合索引、GIST空间索引以及表与表之间“一对多”约束关系链。

[0163] 3、测试评估

[0164] 为确保“地质资料信息集群机理与接口聚合系统”的正确性、稳定性、安全性和可操作性,依据中国地质调查局地质调查技术标准DD2010-01《地质调查软件开发测试管理规程》,2011年7月份组织专家利用专业软件对标定的各项技术指标进行了严格测试,尤其体现在集群节点模拟、并发访问压力测试、底层接口边界条件测试和跨平台兼容性测试等方面。

[0165] 4、部署运行

[0166] 具体实施包括安装、配置、数据制备、节点组网、接口调用五个环节。

[0167] 4.1安装

[0168] 本发明成果以纯软件形式运行于Windows XP/2003/VISTA/2008/7/8操作系统X86/X64架构之上,需额外安装以下免费软件:

[0169] √基础运行环境Microsoft.NetFramework 4及其以上版本;

[0170] √关系型数据库管理系统PostgreSQL 9.1.4及其以上版本;

[0171] √空间数据库引擎PostGIS 2.0及其以上版本;

[0172] 可将软件直接复制到服务器某文件夹下,在Internet信息服务(IIS)管理器中将该文件夹转换为虚拟目录,添加成应用程序并启用ASP.NET v4.0应用程序池。在资源管理器中将该文件夹的访问属性添加[NETWORK SERVICE]账户并赋予[读写控制]权限。

[0173] 4.2配置

[0174] 1)修改web.config文件中的connectionStrings节,其内容是数据库的连接信息,可视具体情况设置数据库名、用户名和密码。

[0175] 2)修改web.config文件appSettings节,以便设置集群信息树顶级节点名称及其描述信息,还可控制接口调用时返回的坐标精度。

[0176] 4.3数据制备

[0177] 方法1:可利用“地学空间信息检索系统-GEOSite”门户网站具备的交互操作功能,手工制作点线面及贴图要素,设置每个要素的属性内容和可视化参数,将其按XML格式;

[0178] 方法2:可将MAPGIS点线面文件转换成SHP格式,在GEOSite工具箱中转换成XML格式;

[0179] 方法3:谷歌地球KML文档可直接用于发布或转换成XML;

[0180] 方法4:可将符合要求的EXCEL格式数据另存为制表符分隔的TXT文件,经GEOSite

工具箱转换成XML;

[0181] 方法5:中国地质调查局地质资料联合编目系统导出的XML格式文件可经GE0site工具箱转换成可发布的XML。

[0182] 所述GE0site工具箱为辅助性工具软件。

[0183] 4.4节点组网

[0184] App_Data文件夹下的register.xml文件是集群节点体系控制文档,系统管理员可在服务器端用记事本修改,还可在浏览器端通过访问register.aspx(需要输入数据库密码并具有写入权限!)进行远程修改。

[0185] 数据格式如下:

[0186] <geo><--根节点名必须是geo并保持唯一-->

[0187] <group><--群组节点名必须是group,可有零个或多个-->

[0188] <address>远程路径;支持http、https、ftp和file协议</address>

[0189] <domain>域名</domain>

[0190] <username>用户名</username>

[0191] <password>密码</password>

[0192] </group>

[0193] </geo>

[0194] 其中<address>节指向某群的访问路径,该路径内存放可供发布的地质资料信息xml、kml或压缩格式gz文档。此类文档的命名规则形如:geosite序号.扩展名

[0195] 其中,[序号]必须从0开始编号,严格保证顺序编号(若不连续,则自动中止后续文件的发现机制)![扩展名]暂支持xml、kml和gz三种命名方式。

[0196] 4.5接口调用

[0197] “地质资料信息集群机理与接口聚合系统”呈现的REST接口指令、调用参数及含义如下:

[0198] help=>显示所有接口函数及描述信息。可用于检查系统安装的正确性;

[0199] refresh?domain={domainname}>采用并行技术“拉、推、送”模式调用数据中心集群架构信息发现、获取、同步、对等、拓扑引擎。Domain参数可为空=发布池、all=整个群、空格分隔的顶级域名列表=某些域;

[0200] domain=>获取已注册的集群架构顶级域名列表;

[0201] optimization=>优化集群数据库性能,尤其针对索引崩溃或数据变更以及主详表不对应时需执行本命令。经过整理后的数据库,将呈现各子表的严格一致性;

[0202] dataware=>返回数据仓节点树;

[0203] alldatasets=>等价于datasets;

[0204] datasets=>返回数据仓下所有不重复的数据集及父代节点树;

[0205] datasetstotal=>返回数据仓下所有不重复的数据集总数;

[0206] dataset?setname={datasetname}>返回数据仓下指定数据集datasetname及父代节点树;

[0207] alldatabases=>获取数据仓下所有数据库及父代节点树;

[0208] databases?setname={datasetname}>返回数据仓下指定数据集datasetname

内所有不重复的数据库及父代节点树；

[0209] `databasestotal?setname={datasetname}` => 返回数据仓下指定数据集 `datasetname` 内所有不重复的数据库总数；

[0210] `database?setname={datasetname}&basename={databasename}` => 返回数据仓下指定数据集 `datasetname` 内指定数据库 `databasename` 及父代节点树；

[0211] `alllayers` => 获取数据仓下所有图层及父代节点树；

[0212] `layers?setname={datasetname}&basename={databasename}` => 返回数据仓下指定数据集 `datasetname` 内指定数据库 `databasename` 里所有不重复的图层及父代节点树；

[0213] `layerstotal?setname={datasetname}&basename={databasename}` => 返回数据仓下指定数据集 `datasetname` 内指定数据库 `databasename` 里所有不重复的图层的数量；

[0214] `layer?setname={datasetname}&basename={databasename}&layname={layername}` => 返回数据仓下指定数据集 `datasetname` 内指定数据库 `databasename` 里指定图层 `layername` 及父代节点树；

[0215] `allelements?page={p}&many={m}` => 获取数据仓下指定页码 (`page >= 0`)、指定数量 (`many >= 0`) 的图元及父代节点树。`page` 为 0 时，忽略 `many` 而直接返回所有图元；`page` 为负值时，返回图元个数；

[0216] `elements?page={p}&many={m}&setname={datasetname}&basename={databasename}&layname={layername}` => 获取数据仓下指定页码 (`page >= 0`)、指定数量 (`many >= 0`) 的指定数据集内指定数据库里指定图层内所有图元及父代节点树。`page` 为 0 时，忽略 `many` 而直接返回符合要求的所有图元；`page` 为负值时，返回图元个数；

[0217] `elementstotal?setname={datasetname}&basename={databasename}&layname={layername}&type={elementtype}` => 返回唯一数据仓下指定数据集 `datasetname` 内指定数据库 `databasename` 里指定图层 `layername` 内指定图元类型的图元数 0=点 1=线 2=面 3=贴图 其他=所有类型；

[0218] `elementid?max={minormax}` => 获取所有图元个数以及其中最小或最大 ID 码 0: 最小 1: 最大 其他: 图元个数；

[0219] `pointwithin?page={p}&many={m}&lng={x}&lat={y}` => 获取指定页码 (`page >= 0`)、指定数量 (`many >= 0`) 的指定坐标落入其内的所有图元及父代节点树。`page` 为 0 时，忽略 `many` 而直接返回符合要求的所有图元；`page` 为负值时，返回图元个数；

[0220] `linecross?page={p}&many={m}&coordinate={linestring}` => 获取指定页码 (`page >= 0`)、指定数量 (`many >= 0`) 的与指定线相交的所有图元及父代节点树。注意：经纬度之间用空格分隔，坐标对之间用逗号分隔！`page` 为 0 时，忽略 `many` 而直接返回符合要求的所有图元；`page` 为负值时，返回图元个数；

[0221] `polygoncontains?page={p}&many={m}&method={intersect}&coordinate={polygon}` => 获取指定页码 (`page >= 0`)、指定数量 (`many >= 0`) 的落入 (`intersect: 0`) 或相交 (`intersect: 非 0`) 指定范围的所有图元及父代节点树。注意：经纬度之间用空格分隔，坐标对之间用逗号分隔！结束点必须等于起始点，以便多边形正确封闭！`page` 为 0 时，忽略 `many` 而直接返回符合要求的所有图元；`page` 为负值时，返回图元个数；

[0222] `spatialanalyse?page={p}&many={m}&method={intersect}&setname=`

{datasetname}&basename={databasename}&layname={layername}=>获取指定页码 (page>=0)、指定数量(many>=0)的与指定数据集内指定数据库里指定图层内的图元呈落入(intersect:0)或相交(intersect:非0)关系的其他图元及父代节点树。page为0时,忽略many而直接返回符合要求的所有图元;page为负值时,返回图元个数;

[0223] spatialanalyse2?page={p}&many={m}&method={intersect}&mothersetname={datasetname}&motherbasename={databasename}&motherlayname={layername}&childsetname={datasetname2}&childbasename={databasename2}&childlayname={layername2}=>获取指定页码(page>=0)、指定数量(many>=0)的指定数据集内指定数据库里指定图层内与指定数据集内指定数据库里指定图层内的图元呈落入(intersect:0)或相交(intersect:非0)关系的其他图元及父代节点树。page为0时,忽略many而直接返回符合要求的所有图元;page为负值时,返回图元个数;

[0224] keysearch?page={p}&many={m}&method={orand}&key={keys}=>获取指定页码(page>=0)、指定数量(many>=0)的经由关键字列表(以空格或逗号为分隔符)可按method:0=>“并且”、method:非0=>“或者”两种运算模式对图元属性内容(忽略字段名称)进行模糊匹配的图元及父代节点树。page为0时,忽略many而直接返回符合要求的所有图元;page为负值时,返回图元个数;

[0225] 空白=>获取集群架构信息。

[0226] 为便于阐明并行化技术与传统遍历的区别,将本发明中“并行反向对等”C#语言实现算法摘录如下:

[0227]

#region 反向对等过程: 对过时时间戳文档进行清理,使得分布式集群架构信息与服务器数据中心保持同步对等。不对等说明数据库内本时间戳所对应的远程文档已经失效(如:某文档被替代;某文档被删除;某文档内容被更新;服务器端集群注册表发生改变)

//从数据库中先得到所有时间戳

```
var timestamps = from sqldataware in SQLReaderX("SELECT mytimestamp FROM dataware;").Descendants("Xcol") select sqldataware.Value;
```

long pstatus = 0;

//备用方案:传统遍历技术 foreach (var timestampX in timestamps)

[0228]

//采用方案：并行对等技术（完成于 2012 年元月 10 日）

Parallel.ForEach<string, long> //在并行 ForEach 循环中使用线程本地变量，必须使用采用两个 type 参数的方法版本；第一个参数指定源元素的类型，第二个参数指定线程本地变量的类型

```
{ timestamps, //第一个输入参数是数据源
```

```
() => 0, //第二个输入参数是对线程本地变量进行初始化的函数。注意：该函数在每个迭代执行时均执行！
```

```
(timestampX, loop, status) => //第三个输入参数是委托类型 Func<T1, T2, T3, TResult>，每个迭代上的并行循环将调用该委托。其中首个输入参数为当前元素 timestampX，第二个为使能够检查循环状态的 ParallelLoopState 变量 loop，用于在迭代间交互信息，末个为线程本地变量 status，注意：该变量生存期仅限于当前迭代内！
```

```
{ //第一层：将数据库中数据仓表内所有时间戳全部遍历，看看是否能在分布式集群节点内每份文档的时间戳依然对等？事实上，每个时间戳对应一份远程 xml 文档
```

```
bool t2t = false; //是否对等的逻辑变量：真=对等 假=不对等
```

```
for (int i = 0; i < get_groups(); i++)
```

```
{ //第二层：遍历注册表字典包含的整个集群编号
```

```
for (int j = 0; j < get_geos(i); j++)
```

```
//第三层：遍历某集群编号下所有文档
```

```
if (timestampX == string.Format("{0}-{1}-{2}", i, j, get_timestamp(i,
```

```
]))
```

```
{ t2t = true; break; }
```

```
if (t2t) break; //若对等便无需做任何事情，退出本层遍历
```

```
}
```

```
if (!t2t) //遍历结束后仍未发现对等，便说明数据库内本时间戳所对应的远程文档已经失效（1、某文档被替代；2、末文档被删除；3、某文档内容被更新；4、服务器端集群注册表发生改变），应该删除本条记录和其子代记录！
```

```
{ bool deleteX = false; //指示是否需要删除记录
```

```
if (addresslinks.Length == 0) //若未指定对等路径，便从数据仓中清理多
```

```
余的时间戳记录。
```

```
deleteX = true;
```

[0229]

```

else//若指定了对等路径，需继续判断决策
    //Regex.Split(timestampX, "-");返回的正则数组中，[0]==群号 [1]==
文档编号 其他暂无需关注
    if (get_geos(int.Parse(Regex.Split(timestampX, "-")[0])) >= 0) //说明
此时间戳对应的群的确与指定的地址所对应的群一致，而时间戳又不对等，因此需要删除
        deleteX = true;
    if (deleteX)
    {
        //并行采用下行
        status = 1;
        //传统遍历采用本行 pstatus++;
        SQLNonQuery(string.Format("DELETE FROM dataware WHERE
mytimestamp='{0}';", timestampX)); //支持级联删除。得益于对关联字段已经建立了主键-外键
删除层叠约束关系，使得仅删除数据仓某记录便可实现子代数据库内记录的删除！
    }
}
else status = 0;
return status; //返回线程本地变量，方法随后会将其传递到此分区上的下一个迭代
}
//最后一个输入参数是 Action<T>委托，当所有循环完成时，方法将调用该委托
//方法为此线程（或循环分区）提供线程本地变量的最终值，提供代码捕获最
终值，并执行将此分区中的结果和其他分区中的结果进行合并所需的任何操作。由于委托类
型为 Action<T>，因此没有返回值
//并行采用下行
(finalResult) => Interlocked.Add(ref pstatus, finalResult) //锁定原子(cpu 寄存器指
令)变量 pstatus，并将由 status 传递给 finalResult 的值进行累加
};
#endregion

```

[0230] 本发明能够解决的技术问题是针对行业内地质信息集中式和分布式馆藏现状，将多元异构空间信息进行统一化描述，依托信息技术前沿理论，通过海量信息发现、获取、转换、存储、同步、对等、聚合、重构、推送等机制，实现基于关键字匹配、类别筛选、空间约束条

件的检索服务,为跨平台、跨行业应用开辟信息传递无障碍通道,向外界呈现具有“推拉接送”集群机理的开放型嵌入式接口;建立具松散耦合(弹性)特征的网状或树状结点体系,为部、局、大区中心、省厅、地市、基层单位乃至个人提供可定制、可延伸、可跨域的多级组网能力,为构建广域覆盖、广泛参与的地质资料信息结点群框架提供运行支撑引擎。

[0231] 以上通过具体的和优选的实施例详细的描述了本发明,但本领域技术人员应该明白,本发明并不局限于以上所述实施例,凡在本发明的精神和原则之内,所作的任何修改、等同替换等,均应包含在本发明的保护范围之内。

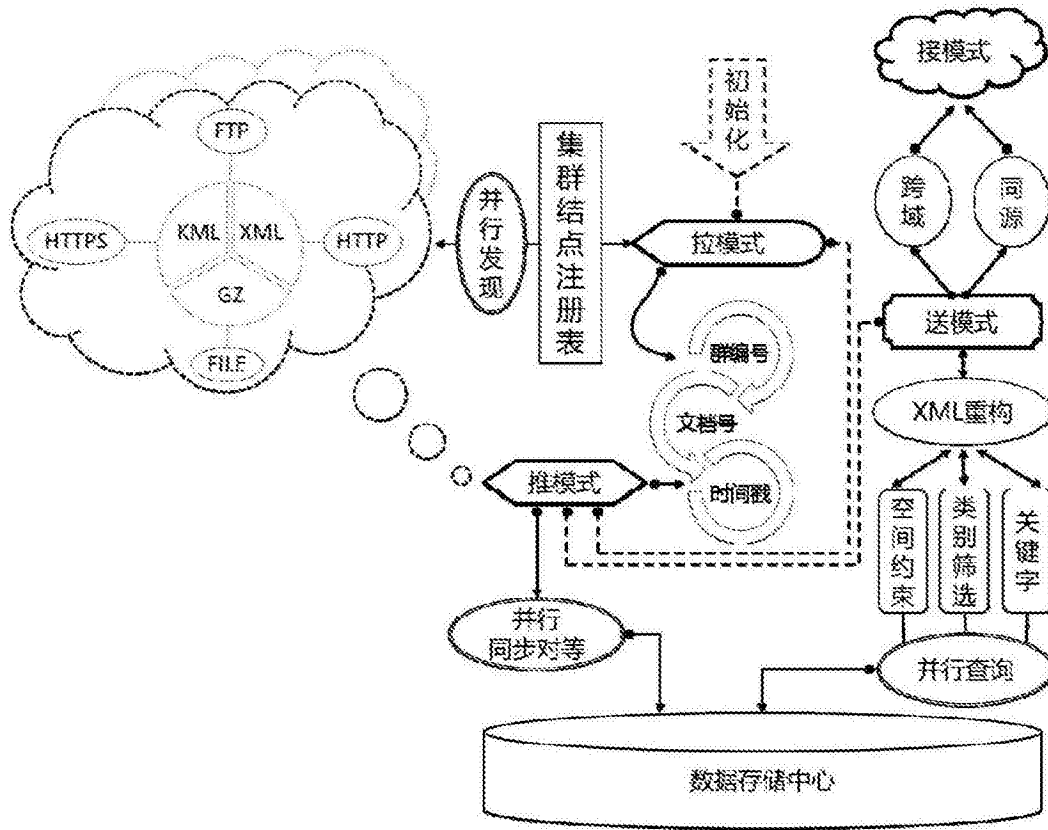


图1

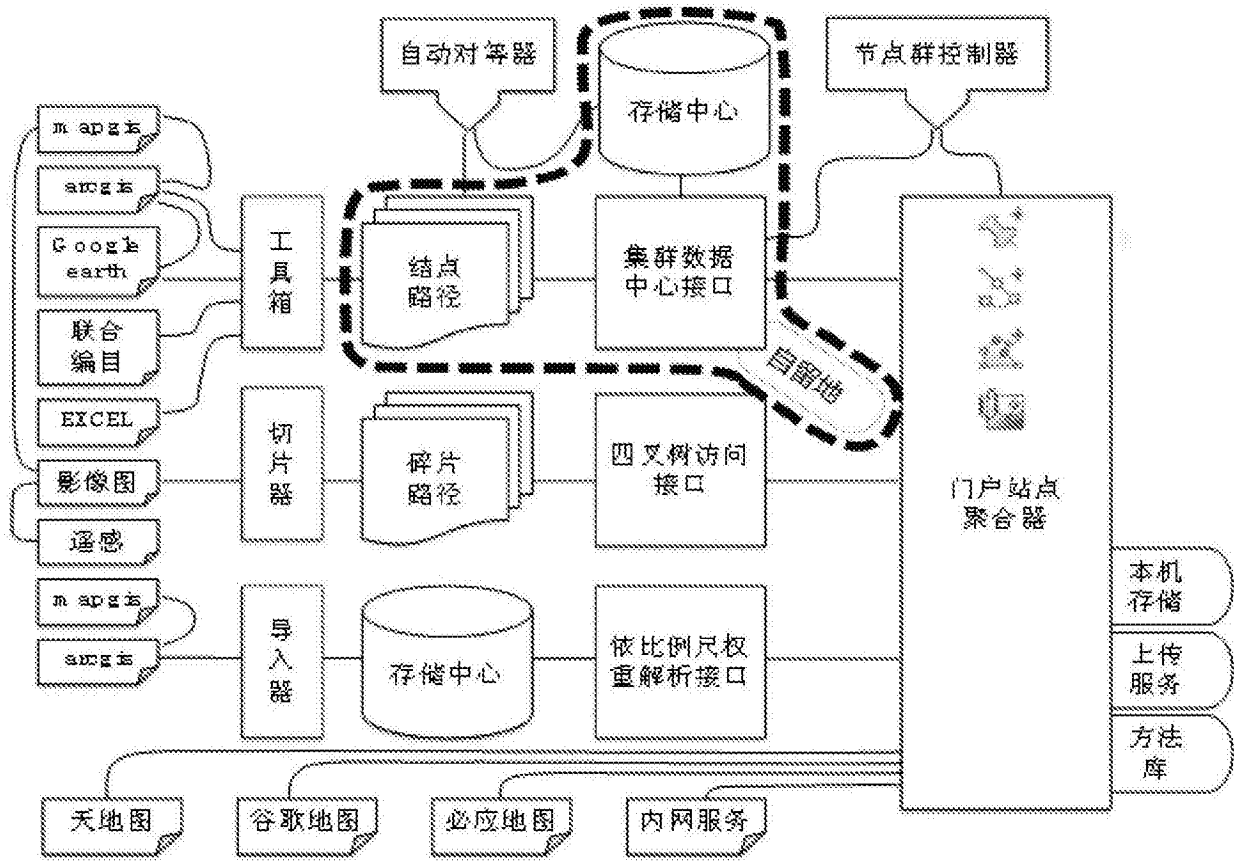


图2

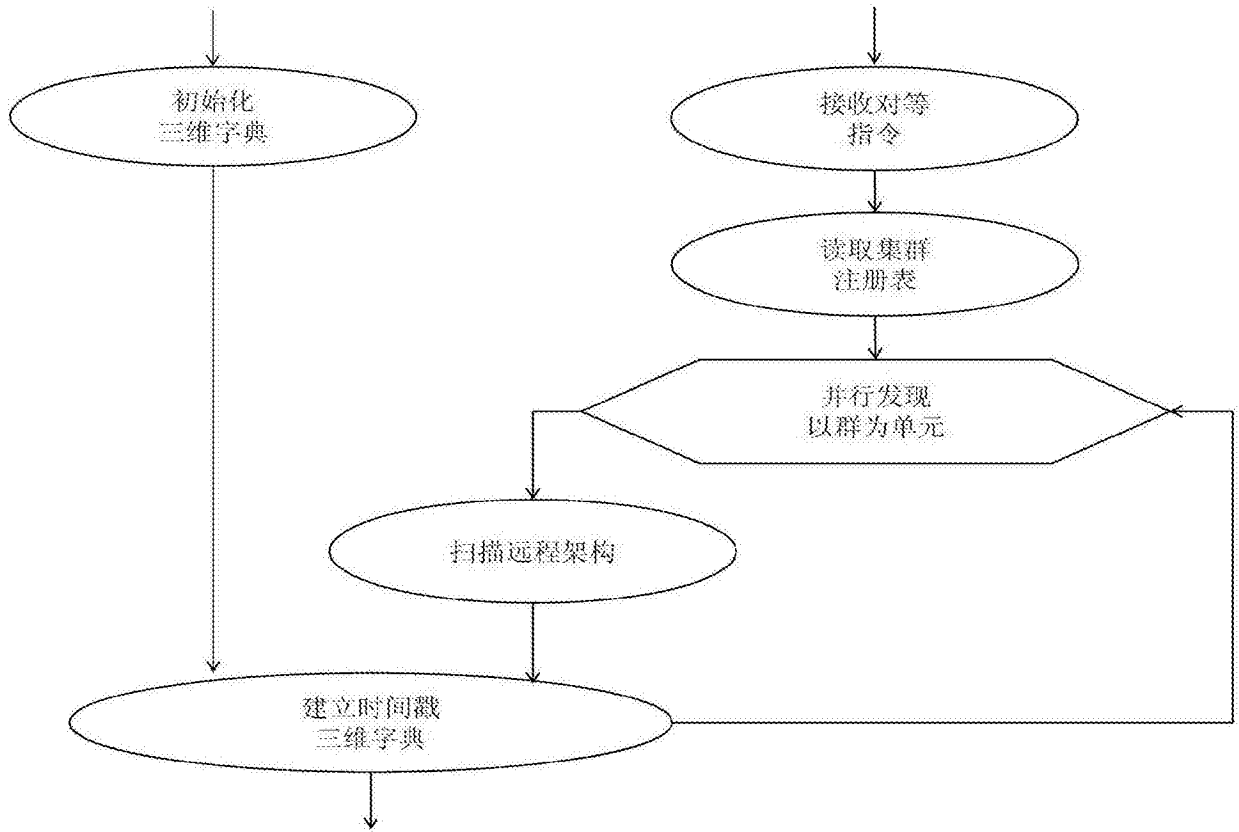


图3

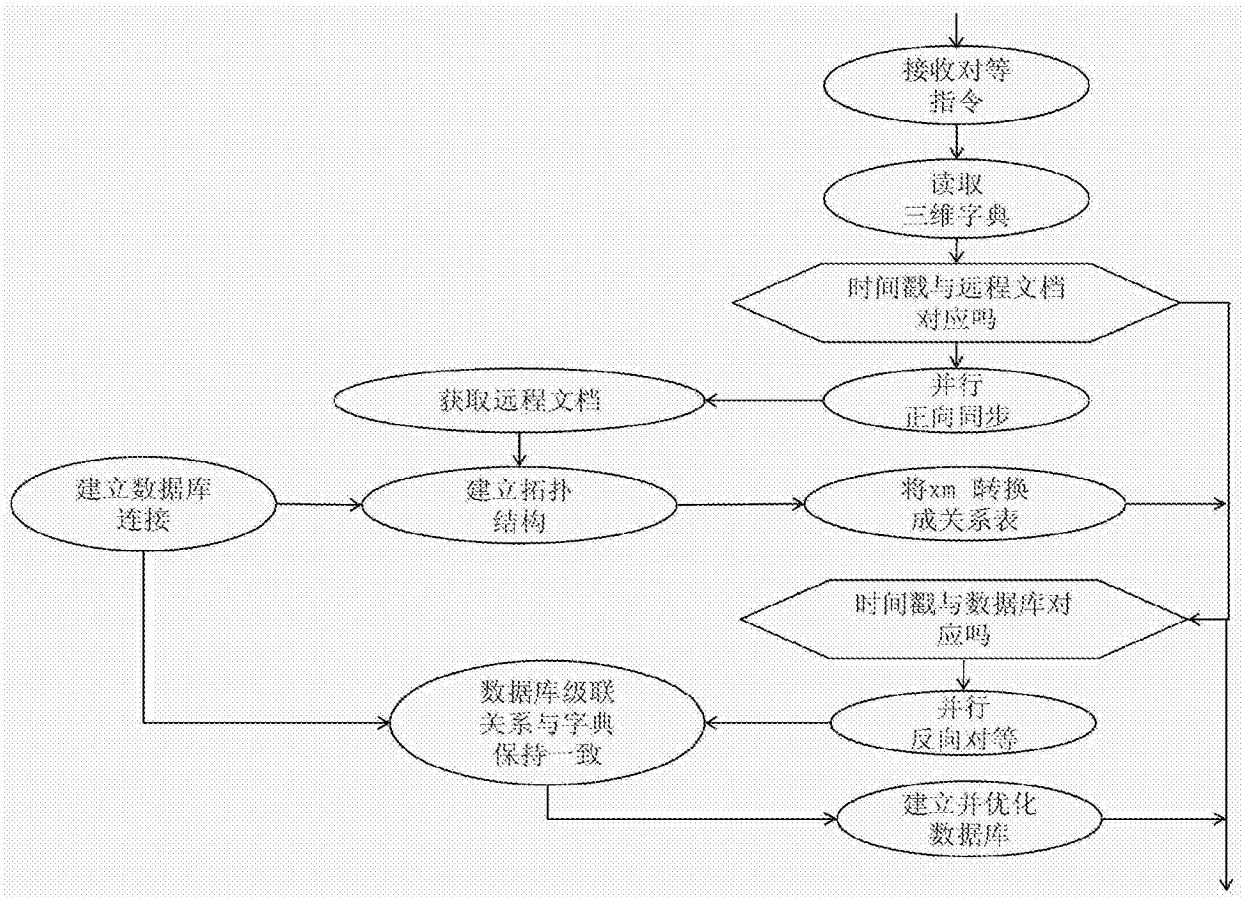


图4

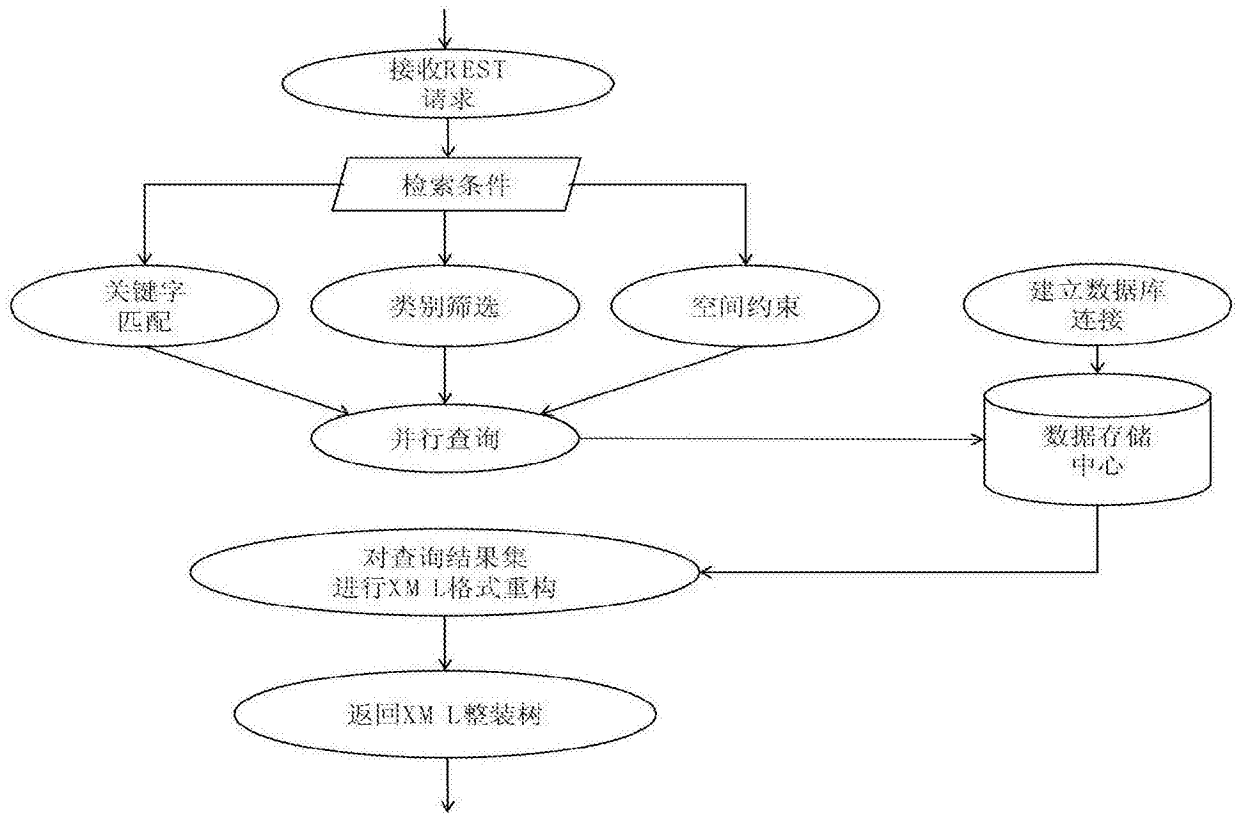


图5

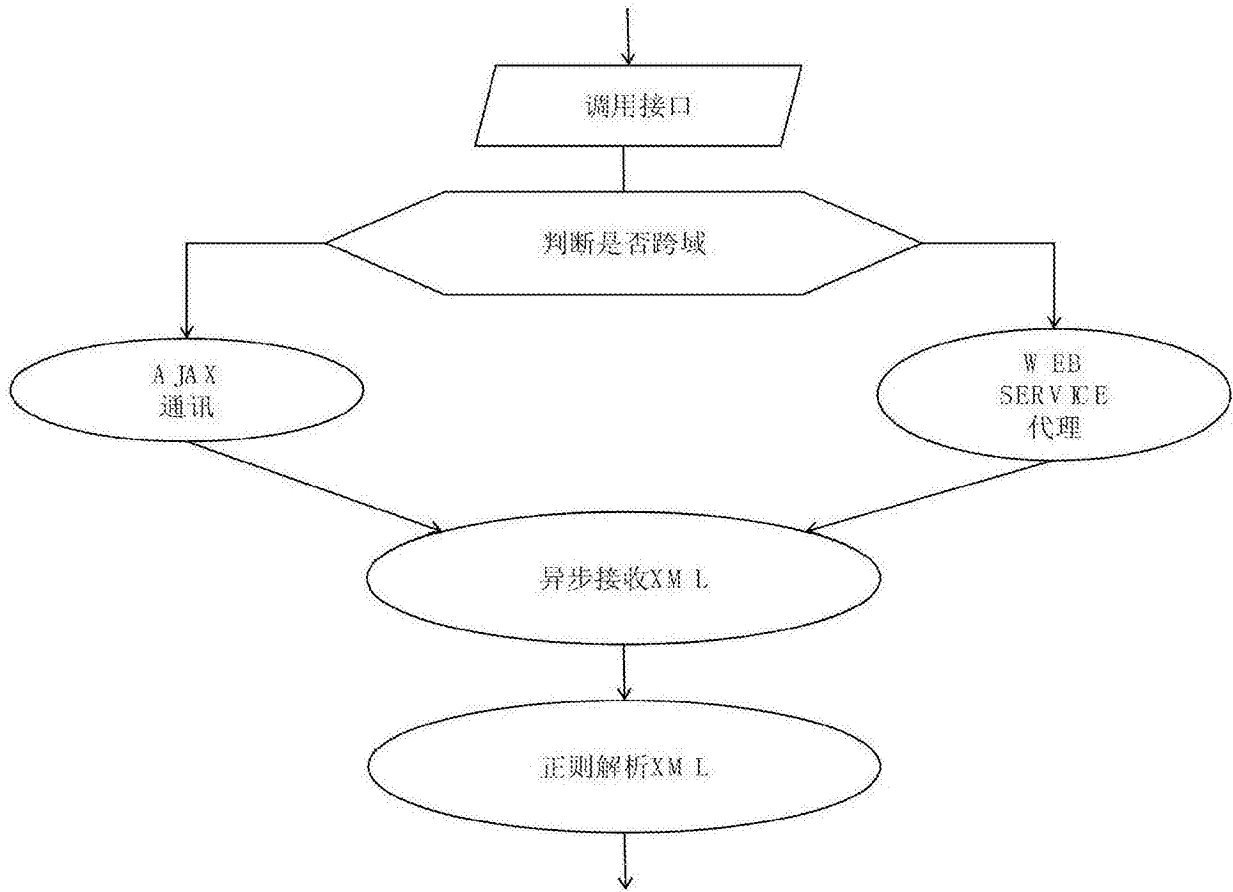


图6

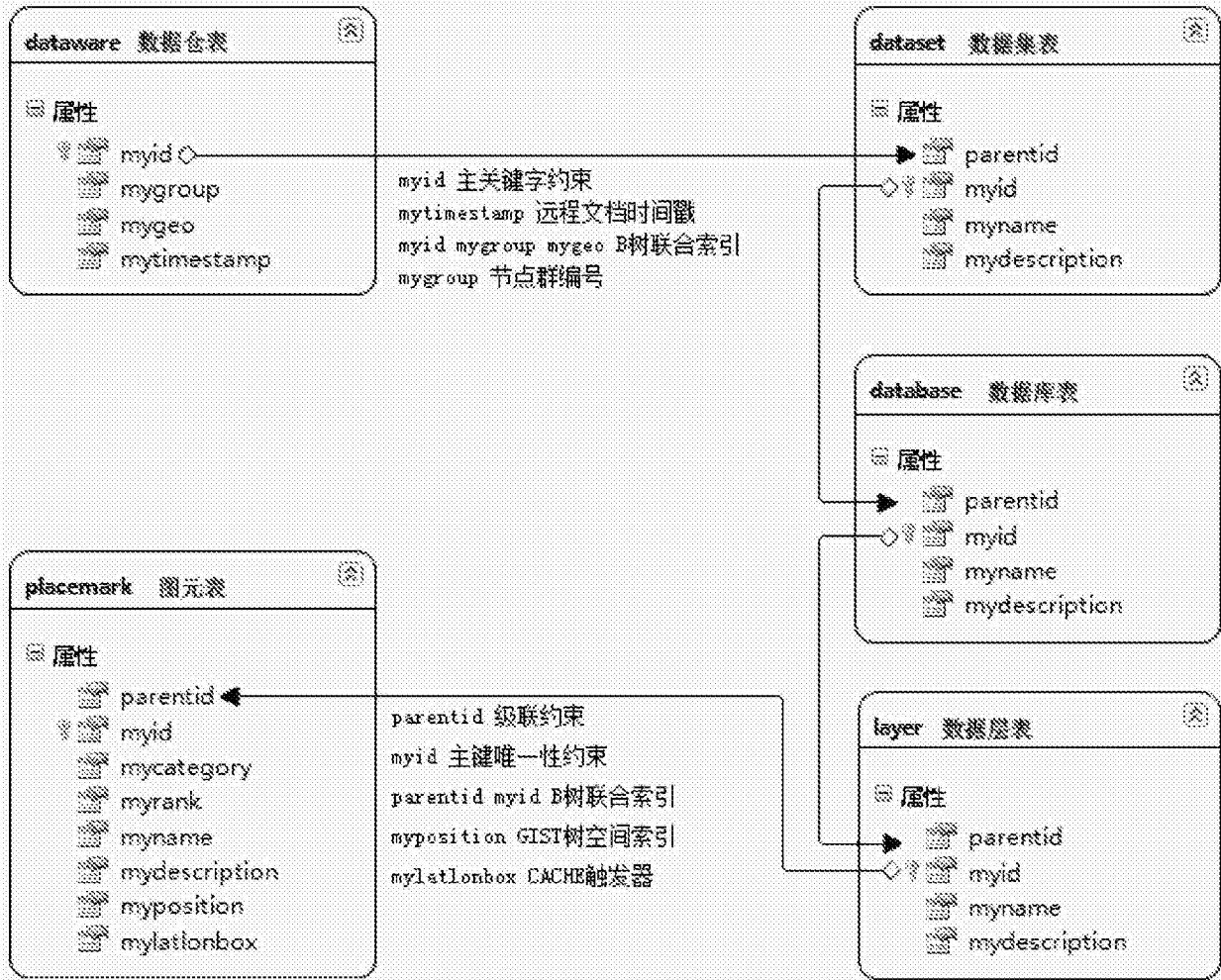


图7