

基于移动 Agent 的分布式地理信息查询

关侏红¹ 陈晓龙¹ 陈俊鹏¹ 周水庚²

(1 武汉大学计算机学院, 武汉市珞喻路 129 号, 430079)

(2 复旦大学计算机科学与工程系, 上海市邯郸路 220 号, 200433)

摘要:提出了基于移动 Agent 的分布式地理信息查询的新思路,给出了构建基于移动 Agent 的分布式地理信息查询系统的方法,并利用移动 Agent 开发平台 Aglets 开发了一个分布式地理信息查询原型系统,实现了分布式地理信息的透明访问。

关键词:地理信息系统; 分布式地理信息查询; 移动 Agent; Aglets; 原型系统

中图法分类号: TP393.4

随着互联网(Internet)的迅猛发展和万维网(WWW)的广泛使用,特别是近年来移动 Internet 的快速出现和移动设备(手机、掌上电脑等)上网用户的飞速增长,对地理信息系统(GIS)及其服务提出了新的需求,即在任何时间、任何地点、以任何方式能够快速、方便地获取所需要的地理信息及其相关信息。

现有基于 Internet 的 GIS 大都是采用传统的 Client/Server (C/S) 或 Browser/Server (B/S) 体系结构。这类系统一般对网络带宽的依赖性很强,而且需要在客户端和服务端之间建立稳定的连接,因此,比较适合于具有较宽的带宽且连接稳定的局域网环境;而在广域、异构、低带宽、信息源分散、连接不稳定的 Internet 环境下,就难以有效地提供分布式地理信息服务。特别是对移动 Internet 而言,移动设备(如手机)和无线通信网的局限性(如内存、计算能力和供电能力有限,通信带宽窄且连接经常中断以及延迟时间长等),使这一问题更为突出。此外,C/S 或 B/S 模式对于客户端的应用不支持直接的分组感知,因而限制了其在高度互操作分布式系统中的应用。

近年来基于移动 Agent 的分布计算技术得到了较快的发展。移动 Agent 的跨平台移动、动态和分布计算等特性扩展了传统 Agent 处理事务的能力,使之能够较好地解决(移动)Internet 环境中低带宽和不稳定连接的问题,并可以提供对网

络中非持续性连接设备(如便携机、掌上电脑、手机)上网查询的支持;而且,移动 Agent 的智能性与灵活性有利于在分布、异构、动态环境下实时有效地提供 GIS 信息服务,并有利于提高系统的稳健性^[1,2]。

移动 Agent 所具有的优势使该技术已经在电子商务、信息检索、网络监控与管理等方面得到应用。本文介绍了利用移动 Agent 实现的一个分布式地理信息查询原型系统。

1 移动 Agent 和移动 Agent 开发平台 Aglets

1.1 移动 Agent

移动 Agent 是 Agent 技术与分布式计算技术相结合的产物。Agent 概念起源于人工智能领域,但随着有关 Agent 的研究在多个领域相互渗透和逐步深入,Agent 概念已经衍生出许多涵义和理解,并在人工智能、分布式系统、软件工程及通信等多个领域中得到应用。基本上达成了这样的共识:Agent 指一种实体,它能够在一定环境下持续自主地完成自身功能,而不需要人的干预或指导;或指在动态的、不可预测的环境中的相对独立的计算系统,能够感知和解释反映环境的事件,并通过执行动作来对环境产生影响。

移动 Agent 分布式计算是在早期的一些分布

计算模式如远程过程调用(remote procedure call, RPC)、远程计算(remote evaluation, REV)的基础上发展演变得到的^[3]。它们之间的区别在于:在远程过程调用中,数据在客户和服务端之间双向传送;在远程计算中,代码从客户发送到服务器,数据返回;而一个移动 Agent 是包含代码、数据和执行情景的一个程序,由客户发送到服务器,移动 Agent 按照其任务可以动态自主地在网络中的多个服务器之间迁移,在合适的情况下带着结果迁移回客户端,因此具有更大的自主性、灵活性。

基于移动 Agent 的分布式计算模式从提出到现在一直是研究焦点。从第一个明确表示为使用移动 Agent 进行编程而设计的系统 Telescript 到现在,出现了 Tacoma、Agent Tcl、Aglets、Voyager、Concordia、Ajanta 等多种移动 Agent 实验性系统,这些移动 Agent 系统往往作为开发基于移动 Agent 的应用系统的平台,其中最具有代表性的当属 IBM 公司开发的 Aglets。

1.2 移动 Agent 开发平台 Aglets

Aglets 是一个基于 Java 的移动 Agent 平台^[4]。Aglets 提供的移动 Agent 编程模型即 Client/Agent/Server 的结构比传统的 C/S 结构在系统的执行性能上有了很大的提高,在无线、拨号网络中其性能提高将近 10 倍,在固定网络中性能提高也达到了 30%~40%^[5]。

在 Aglets 系统中,Aglet 就表示移动 Agent。Aglet 产生于一台主机上,并可以在不同的主机之间迁移,不需占用太多的系统资源和网络带宽,因此适宜于低带宽和不稳定连接的 Internet 环境。通过规划 Aglets 的巡回路线(itinerary),使得 Aglets 有能力在网络中按照巡回路线灵活地进行迁移,并通过 Aglets 间的通信方法实现移动 Agents 之间的协作,从而可以从多个节点收集所需

数据。

Aglets 运行环境由一个被称为 Tahiti 的 Aglet 服务器提供。Tahiti 提供移动 Agent 产生、(自我)复制、派遣和销毁等基本功能。

Aglets 的基本功能和特性由三个类来定义:Java Aglet、AgletProxy 和 AgletContext。其中,抽象类 Aglet 定义用于控制移动 Agent(即 Aglet)移动和生命周期的各种基本方法;同时,提供对 Aglet 内部属性(包括 Aglet 的产生时间、所有者、代码库、信任级别等)和动态属性(如 Aglet 到达某一站点的时间和当前情境的地址)的访问。AgletProxy 类提供访问 Aglet 的句柄(handle)。Aglet 通过 AgletProxy 对象把需求转发给远地主机并将结果返回给本地主机,从而实现对 Aglet 的地址透明。AgletProxy 的另一个作用是屏蔽其他对象对 Aglet 对象的直接访问。Aglets 之间的所有通信都是通过 AgletProxy 对象进行。AgletContext 类提供移动 Agent 在 Tahiti 服务器内的运行环境。当一个 Aglet 要迁移到某一远地站点时,首先和当前的 AgletContext 对象脱离关系,然后将自身代码、数据和状态序列转化成消息字节流(Message bytestream),通过网络传到远地站点后在当地的 AgletContext 内重建自己,并在当地 AgletContext 中执行自己未完的使命。图 1 说明了 Aglet 在两个站点间的迁移过程。A、B 两个站点都有一个静态 Agent:Aglet-A 和 Aglet-B。Aglet-A 产生移动 Agent:Aglet-C,并获得 Aglet-C 的代理引用。同时,Aglet-A 也将自己的代理引用传给 Aglet-C。这样,Aglet-A 和 Aglet-C 之间就可以交换消息。Aglet-C 迁移到 B 站点后,也和 Aglet-B 交换代理引用。这时,Aglet-C 就可以引用 Aglet-A 和 Aglet-B 的方法和数据。

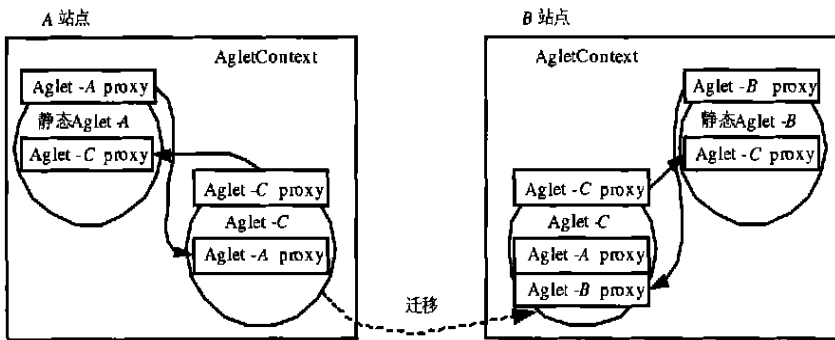


图 1 Aglet 的迁移过程

Fig. 1 Aglet's Migration

2 利用 Aglets 查询分布式地理信息

2.1 系统框架

一个基于 Aglets 的分布式地理信息查询系统主要由 4 个部分构成, 即服务器、客户机、连接这些结点的互联网或内部网络以及漫游于网络中往来于各个结点之间并代表用户执行信息访问与检索任务的移动 Agent 即 Aglets。系统框架如图

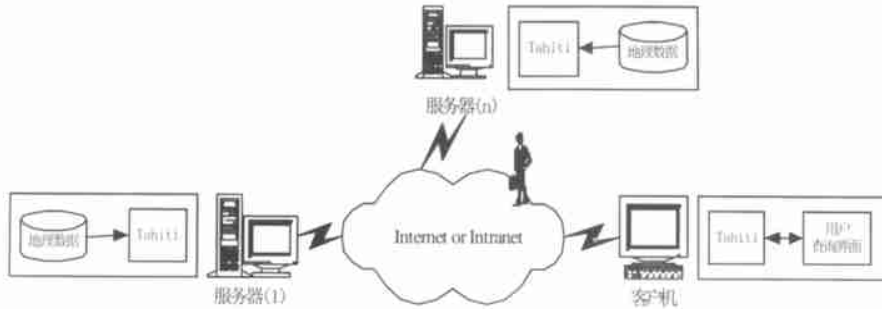


图 2 基于 Aglet 的分布式地理信息查询系统框架

Fig. 2 Framework of Distributed Geographic Information Query System Based on Aglet

在基于 Aglets 的分布式地理信息查询系统中, 可以将地理信息的相关数据分布于网络中的多个服务器上, 在服务器端启动 Aglets 的 Tahiti 服务器端执行环境, 所有从客户端发至服务器的 Aglet 线程都在 Tahiti 执行环境下运行。客户端需要查询地理信息时, 其处理过程是通过查询界面提交查询请求, 系统中分布地理信息查询的处理流程如下。

1) 用户通过查询界面提交查询请求。

2) 客户端的 Tahiti 服务器有一个静态 Aglet 用于接收用户查询。静态 Aglet 接收到用户查询后, 对用户查询进行分析, 并根据用户查询涉及到的数据源制定移动 Aglets (负责到相关服务器端获取地理信息) 的巡回查询计划。

3) 静态 Aglet 生成移动 Aglets, 并将查询计划交付给移动 Aglets。

4) 移动 Aglets 携带查询计划, 迁移到目标服务器, 获取所有相关地理数据, 然后返回客户端。

5) 查询结果在客户端显示, 此时, 用户可以对查询结果进行交互式操作。

其中, 移动 Aglets 的巡回路线规划问题由服务器完成, 按照用户提交的查询请求解析查询任务, 并由服务器中的数据资源描述信息、网络感知信息、黄页服务信息等对查询生成相应的子查询计划, 即规划资源访问序列, 找寻最佳巡回路线^[6]。

2 所示。服务器主要由两部分组成: 移动 Agent 服务器 Tahiti 和地理信息数据库; 客户机包括一个移动 Agent 服务器 Tahiti 和一个用户查询与地图显示界面以及地图处理与显示模块。用户看到的只是查询界面, 通过查询界面提交查询请求并显示查询结果, 而对于用户查询请求任务的解析和处理则由后台运行的移动 Agent 服务器 Tahiti 负责。每个 Tahiti 中都有一个静态 Aglet 作为本地 Tahiti 的主机(host)处理一些本地事务。

2.2 系统实现

2.2.1 Tahiti 服务器端执行环境的建立

每个服务器都需要安装 Tahiti, 由此提供 Aglet 源程序运行的后台执行环境。Tahiti 服务器端执行环境的配置信息存放在文件 aglets.props 中; 而 Tahiti 的安全配置则在文件 F:\Documents and Settings\Administrator\aglets\security\aglets.policy 中 (这里设计计算机的操作系统为 Windows 2000, 且操作系统和 Aglets 开发环境均安装在 F 盘上)。如果需要设置某一文件路径以及文件的存取权限, 先启动 Tahiti, 然后在字符界面中的 F:\aglet20\aglets\bin 文件目录下执行如下命令: agletsd -f.\cnf\aglets.props。

2.2.2 客户端程序设计

在构建基于 Aglets 的分布地理信息查询系统时, 在客户端设计了 Aglet 的宿主源程序 gisaglet 和用于启动 Tahiti 执行环境以及显示地理信息的 ShowDialog 程序。gisaglet 线程引发一系列的操作, 如处理地理信息的查询以及启动负责显示的 ShowDialog 源程序等。在这里 gisaglet 作为系统中的宿主程序, 其他的 aglet 源程序则由它派遣出去执行具体的查询任务。

客户端程序需要对查询所需信息的位置进行判断。例如, 当要查询有关湖泊的地理信息时, 只要通过查询界面选择湖泊, 就可以启动后台的 Aglet 程序。Aglet 程序首先判断待查询的湖泊图

层信息是否存放在本地,如果在本地,则可以根据查询条件直接显示查询结果;如果所需的地理信息不在本地,则后台的 Aglet 程序派遣一个 Aglet (假设为 HelloAglet 程序)到相关服务器上执行采集湖泊图层数据的功能。数据访问成功后,HelloAglet 自动返回客户端,然后它可以和 Aglet 宿主程序 gisaglet 进行通讯,随后 HelloAglet 在任务完成后被销毁,则一次查询任务结束。

2.2.3 地图显示

在基于 Aglets 的分布式地理信息查询系统中,为了实现地图显示,利用了 GeoTools 0.8.0 软件包^[7]。GeoTools 是用于地图显示与操作的开放 Java 类库,其处理的主要地理信息文件格式是 ESRI Shapefile。GeoTools 主要用于在 Web 浏览器上开发交互地理信息,在客户端使用 Applet 处理地理图形数据,不需要服务器端的支持。移动 Agent 按照服务器为其规划的巡回路线,从分布在网络中的所需资源所在的站点获取相应的地理信息查询数据。为了提高数据传输效率,查询结果由移动 Agent 从远地站点传回时,采用压缩文件形式,传到客户端后再解开数据文件并进行地图显示。

下面给出的是利用 GeoTools 显示地理图层的一个简单程序:

```
Show Map()
{ Viewer view = new Viewer();
  // Geotools 提供的一个方法,显示类
  URL url = new URL("file:/F:/aglet20/
  aglets/bin/rivers.zip");
  // 绝对路径+地理信息图层文件
  ShapefileReader sfr= new ShapefileReader
  (url);
  // 得到一个读对象;
  t= sfr.getTheme();
  // 得到一个图层对象,赋予一个变量;
  view.addTheme(t);
  // 在视图对象里面添加图层对象,其中图
  层对象对应一个图层数据;
}
```

上述例子给出的只是显示地理图层时必需的操作,如果要对图层中的内容进行查询和特别的表现,则需要做进一步的工作。例如在原型系统中,为了了解各类人种在美国各个州的分布情况,用不同的颜色(用从黄色到红色的变化表示人口由少到多的转变)表示人数的多少(如图 4 所示),这样可以产生更直观的效果。此外,当查询涉及

多个图层的叠加时,则还需要考虑每一图层的显示顺序,以保证多个图层叠加后,不互相遮盖而丢失需要显示的信息。在图 5 中,州这个图层在其他图层添加以后自动重绘,并且在各图层的相对位置中,它处于最下面一层,这样就可以和别的图层叠加后形成一幅完整的地图。另外,通过程序设计可以实现属性数据查询和对地图进行基本操作等功能。

3 一个原型系统

由于 Aglets 是基于 Java 的 Agent 开发平台,为了与其兼容,这里也使用 Java 实现原型系统。使用的 Java 开发软件包括 ASDK (aglets software development kit)2.0 和 JDK 1.3。原型系统是在 Windows2000 下开发的。硬件平台为连接在局域网上的 3 台不同 PC 机,PC 机的配置为 P II-800M、64M 内存和 10G 硬盘。

原型系统中的地理数据取自 ArcView GIS 3.2 中的实例。用的是其中的美国地理数据,包括美国州、主要城市、河流、公路和湖泊等 6 个方面的特征数据;每个特征的数据由 3 个文件构成,分别是地理特征数据文件 (shp 文件)、地理特征数据索引文件 (shx 文件)和地理特征属性数据表 (dbf 文件)。这些数据文件列于表 1 中,它们分别存储在两个服务器站点上;其中州和主要城市数据存放在同一个服务器站点上,其他数据存储在另一个服务器站点上。

表 1 原型系统中的地理数据
Tab. 1 Geographic Data for the Prototype

地理特征	数据文件名	数据文件大小 (bytes)	数据存储站点
州	States. dbf	22 423	服务器站点 2
	States. shp	222 344	
	States. shx	508	
	Cities. dbf	1 424 854	
主要城市	Cities. shp	88 272	服务器站点 2
	Cities. shx	25 292	
公路	Roads. dbf	81 738	服务器站点 2
	Roads. shp	252 188	
	Roads. shx	5 532	
湖泊	Lakes. dbf	1 200	服务器站点 2
	Lakes. shp	24 200	
	Lakes. shx	332	
河流	Rivers. dbf	4 634	服务器站点 2
	Rivers. shp	106 988	
	Rivers. shx	548	

图 3 为原型系统的用户查询与地图显示界面

示意图。整个界面分 3 个部分: 查询界面、地图操作区和地图显示区。上面的面板条为查询界面; 中间为地图显示区; 下面的面板条为地图操作区。目前, 地图操作实现了地图缩放、漫游、显示比例设置和恢复初始状态四种功能; 查询界面提供图层选择和针对图层的属性约束查询。用户只需用鼠标点击相应的复选框即可以查询所需要的图层。层图属性约束查询目前仅实现了对州层图的查询。层图属性约束查询界面由 3 个下拉框构成: 州层图属性框、操作框和参数设置框。属性框用于选择州层图的属性, 如人口数、面积等; 操作框用于选择操作, 目前设置了三种比较操作, 即“大于”、“小于”和“等于”; 参数设置框用于设置具体参数。预定了 MAX 和 MIN 两个缺省参数, 分别表示所选属性值的最大值与最小值。当然, 这只有在选择了具有数字属性时才适用。设置这两个缺省参数, 是为了表达像“查找人口(或面积)最大(小)的州”之类的用户查询。图 4 和图 5 给出了两个具体的查询实例, 图中不包括美国阿拉斯加和夏威夷群岛。

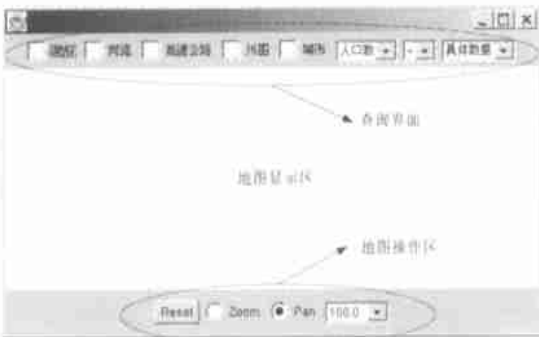


图 3 原型系统的用户查询与地图显示界面
Fig. 3 Prototype Interface for User Query and Map Display

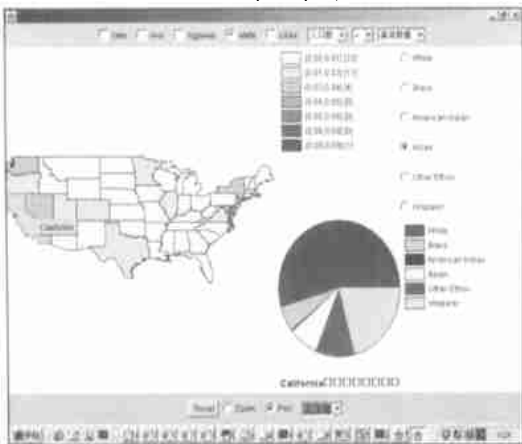


图 4 美国各州人口查询结果

Fig. 5 Query Result of the American States' Population



图 5 美国州、公路、河流和湖泊查询结果

Fig. 5 Query Result of the American States' Highways, Rivers and Lakes

4 结 语

本文介绍了一个利用移动 Agent 访问分布式 GIS 的原型系统及其相关实现方法与技术。该系统是在 IBM 的移动 Agent 平台 Aglets 的基础上建立的, 目前包括两个服务器站点。地理空间数据分布在这两个站点上, 用户通过第三方客户机可以透明访问分布在两个服务器站点上的数据。原型系统提供图形查询界面、地图显示界面和简单的地图操作功能。该系统的成功运行表明, 利用移动 Agent 访问分布式 GIS 是可行、有效的。为了更加有效地实现针对分布式地理信息的访问, 在今后的研究工作中, 将采用 GML 技术来处理分布地理信息的异构性; 另一方面, 探索利用移动 Agent 实现移动 GIS 服务也是今后研究的一个重要问题。

参 考 文 献

- 1 Kotz D, Gray R S. Mobile Agents and the Future of the Internet. ACM Operating Systems Review, 1999, 33(3): 7~13
- 2 Lange D B, Oshima M. Seven Good Reasons for Mobile Agents. Communications of the ACM, 1999, 42(3): 88~89
- 3 Tripathi A R, Karnik N M, Vora M K, et al. Ajanta-A System for Mobile Agent Programming. TR98-016, Minnesota, 1998
- 4 Aglets URL, <http://www.trl.ibm.co.jp/aglets> 2002
- 5 Papastavrou S, Samaras G, Pitoura E. Mobile Agents for World Wide Web Distributed Database Access. IEEE Transactions on Knowledge and Data Engineering 2000, 12(5): 802~820

- 6 关侏红. 基于移动 Agent 的分布式 Web GIS 研究: [博士论文]. 武汉: 武汉大学, 2002
- 7 Geotools URL <http://sourceforge.net/project/>, 2002

第一作者简介: 关侏红, 副教授, 研究方向: 分布式 GIS、空间数据库、分布式计算等。发表论文 30 余篇。
E-mail: jhguan@wtusm.edu.cn

Querying Distributed Geographic Information Based on Mobile Agent

GUAN Jihong¹ CHEN Xiaolong¹ CHEN Junpeng¹ ZHOU Shuigeng²

(1 School of Computer, Wuhan University, 129 Luoyu Road, Wuhan, China, 430079)

(2 Department of Computer Science and Technology, Fudan University, 220 Handan Road, Shanghai, China 200433)

Abstract: As a recently developed paradigm in distributed computing, mobile agent can especially benefit (mobile) Internet applications on providing asynchronous task execution and more dynamics, supporting flexible and extensible cooperation, reducing communication bandwidth, enhancing real time abilities and higher degree of robustness, enabling off-line processing and disconnected operation. This paper presents an approach based on mobile agent to query distributed geographic information, which can overcome the drawbacks of the traditional connection-based access method. A prototype is implemented to access distributed geographic information transparently by using Aglets, which is a free mobile agent development platform developed by IBM. In our on-going research project, mobile agents are responsible for remote information retrieval. A user submits spatial or nonspatial query through a GUI interface built at the client machine, and the query is analyzed and decomposed at a server site. Then some mobile agents are created and dispatched to the respective sites to retrieve data autonomously and separately. Distributed GIS based on mobile agent is more adaptable to the open, heterogeneous autonomous (mobile) Internet with limited bandwidth and unstable communication connectivity.

Key words: GIS; distributed geographic information query; mobile agent (MA); Aglets; prototype

About the first author: GUAN Jihong, associate professor. Her research orientations include distributed GIS spatial database, distributed computing, etc. She has published more than 30 papers.

E-mail: jhguan@wtusm.edu.cn