

服务质量敏感型空间信息服务引擎设计

宋正阳¹, 谢传节², 胡玉清³

- (1. 中国矿业大学(北京)机电与信息工程学院, 北京 100083;
2. 地理科学与资源研究所资源与环境信息系统国家重点实验室, 北京 100101;
3. 空军航空管制系, 北京 100071)

摘要: 针对服务优先级不同、服务质量不同、最大限度满足服务质量要求的空间信息引擎, 提出一个新的空间信息服务引擎架构, 该引擎能识别服务请求的上下文信息, 运用不同的服务请求堆栈管理策略和服务请求分发策略, 以满足当服务请求优先级不同时, 优先级高的服务优先得到服务; 当同样优先级时, 服务请求类型服务质量要求高的请求优先得到服务。

关键词: 负载均衡; 服务质量; Web 地理信息系统; 空间信息服务引擎; 基于位置服务

Design of QoS-aware Spatial Information Service Engine

SONG Zheng-yang¹, XIE Chuan-jie², HU Yu-qing³

- (1. School of Mechanical Electronic and Information Engineering, China University of Mining & Technology(Beijing), Beijing 100083; 2. State Key Lab. of Resources and Environmental Information Systems, Institute of Geographical Sciences and Natural Resources Research, Beijing 100101;
3. Department of Air Traffic Control, Air Force, Beijing 100071)

【Abstract】 In order to develop spatial information service engine, which can try its best to meet the Quality-of-Service(QoS) of service requirements, a new Spatial Information Service Engine(SISE) framework is designed. In SISE, service requirements contexts are identified and different policies are used in management of service requirement queues and in the service requirements distribution. In order to meet the requirements of higher priority service, the higher QoS service requirement must be served at first.

【Key words】 load balancing; Quality-of-Service(QoS); Web GIS; Spatial Information Service Engine(SISE); location-based service

1 概述

空间信息作为一种特殊专业信息, 长期以来都由专业系统使用。随着信息技术的发展, 这种状况发生了根本性的变化, 空间信息及其应用无所不在, 而且得到了 IT 主流数据库厂商和软件厂商的广泛支持。目前 Web 下空间信息的应用已非常普及^[1], 其分析和可视化已成为电子商务系统的基本技术。在 Web 下应用空间信息离不开空间信息服务引擎的支持, 如 ArcIMS, MapXTreme 等。但由于用户访问的不确定性, 当前的空间信息服务引擎还不能保障空间信息服务的质量^[2]。

在 Web 条件下, 用户在应用空间信息服务时, 不同情况下的目的不同, 可能是一般性的信息浏览, 也可能是重要的商务流程中的关键环节。显然, 对处在一般浏览模式下, 空间信息服务的质量可以降低要求, 但应尽可能保障处在商务流程中的关键环节的空间信息服务的质量。因此, 就需要空间信息服务引擎具有提供不同质量要求的空间信息服务的能力。

无线网络技术的发展拓展了空间信息的访问方式和访问这些信息服务的设备类型, 如 PDA, 智能终端手机, 可以通过无线网络获取当前位置, 访问空间信息内容, 目前这一类型空间信息服务称为基于位置服务(LBS)。无线用户与传统的空间 Internet 用户使用空间信息服务的不同之处, 是无线用户能等待的时间更短, 这就要求服务有更高的执行和反馈速度, 具有更高的服务质量^[3]。这要求空间信息服务引擎能根据用户连接的不同, 提供不同质量的空间信息服务。

2 空间信息服务的上下文

质量敏感型空间信息服务引擎工作, 首先需要识别服务所要求的服务质量, 而服务质量是由空间服务请求所处的场合、所使用的网络连接和终端设备来识别, 这些信息可称为空间信息服务的上下文(Service Contexts)^[4], 这些上下文信息由服务发出请求时进行指定。

质量敏感型空间信息引擎工作时, 首先通过识别服务请求所携带的上下文信息, 判断服务请求所要求的服务质量, 然后通过综合利用多种策略来对服务请求进行管理, 达到最大限度满足服务质量的需要。

本文服务引擎支持服务的上下文信息, 即在发出服务请求时可明确的上下文包含 3 方面的内容:

(1) 服务的优先级, 可以在服务调用时, 申明服务调用是普通服务请求或是优先服务请求, 如当空间信息服务处在商务流程的关键环节时, 就可以申明服务调用是优先服务调用。

(2) 调用服务的网络连接情景, 网络连接情景可分为 2 种类型: 1) 用 PC 通过网络连接访问空间信息服务; 2) 使用的智能终端, 如手机, 通过无线网络访问空间信息服务, 一般要

基金项目: 国家自然科学基金资助项目“中国西部环与生态科学数据中心西部计划”(90502010); 国家“863”计划基金资助项目(2007AA12Z203)

作者简介: 宋正阳(1984 -), 男, 硕士研究生, 主研方向: 三维数据编码与传输; 谢传节, 副研究员; 胡玉清, 硕士

收稿日期: 2008-12-05 **E-mail:** soaring_eagle@tom.com

求处在无线网络访问下的空间信息服务具有较快的反应速度。

(3)服务的种类,不同的服务要求的服务质量可能会有所不同,如处在无线网络条件下,用户调用位置查询服务时,要求该服务具有更高的服务质量。

3 质量敏感型空间信息服务引擎

质量敏感型空间信息服务引擎由 2 个部分组成,即空间信息服务服务器(Spatial Information Service Server)和服务处理节点(Service Processing Node)。在一个典型引擎部署中,可有一个空间信息服务服务器和多个服务处理节点。引擎结构如图 1 所示。

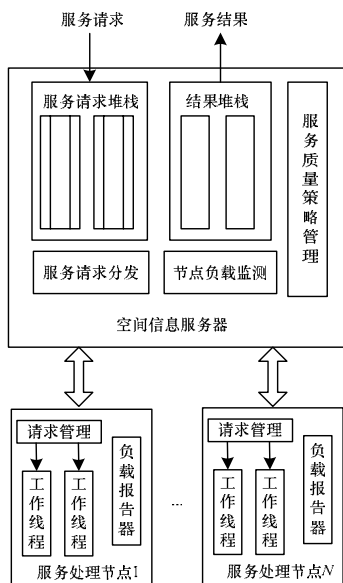


图 1 质量敏感型空间信息服务引擎结构

引擎的空间信息服务服务器由 5 个部分组成:服务请求堆栈,服务结果堆栈,服务请求分发软件,服务处理节点负载监测软件和服务质量策略管理器。

空间信息服务服务器负责接收用户的服务请求,然后对接收的服务请求进行解析,获取服务请求的优先级、服务的类型和所要求返回结果类型,然后将服务请求压入对应的堆栈中。服务请求的优先级有 2 种,即普通服务请求和优先服务请求,这 2 种优先级分别由不同的堆栈进行管理,另外,对于任何一种优先级堆栈,又按具不同服务请求类型,对应不同的子堆栈,以表明不同类型服务请求具有不同的时间紧迫性要求,如在 LBS 应用中,用户当前位置查询服务请求比一般的制图服务请求时间要求更紧迫。

堆栈中的服务请求按一定的策略被选取,交由服务请求分发软件处理。服务请求分发软件根据节点负载信息即各个节点服务请求执行的历史信息,按一定的策略将服务请求分发给某个服务处理节点进行处理。

服务处理节点返回的结果首先被压入到服务器的结果堆栈中,然后结果被按请求的优先级选出,交由结果处理程序处理,结果处理程序再按请求所要求的返回结果类型进行结果转换,并将转化后的结果返回给用户。

引擎服务器中节点负载监测器负责接收服务处理节点定期发回的节点负载信息报告,必要时也可向指定节点发送负载状态信息查询请求,获取该节点的负载信息。

引擎服务器中服务质量策略管理器主要有 2 个方面的任务:(1)调控服务器中的服务质量策略,如请求从请求堆栈中

取出服务请求可有多种策略,这些策略可有策略管理器进行预设,以供请求堆栈在运行中采用。(2)一些管理工作,包括引擎中服务处理节点的管理和服务器中多个可独立运行软件模块的运行监控。

服务处理节点由 3 个部分组成:请求管理软件,工作线程和节点负载状态报告器。请求管理软件负责接收从引擎服务器发送来的服务请求,并调用工作线程来执行服务;工作线程负责服务请求的实际执行,执行结果直接发送到引擎服务器的结果堆栈;节点负载报告器定期向引擎服务器报告该节点的负载信息,另外也响应引擎服务器的负载状态查询。

4 服务质量的策略管理

质量敏感型空间信息服务引擎设计的关键是针对服务质量的各种策略,这些策略包括 2 个方面:服务请求堆栈管理策略和服务请求分发策略。

4.1 服务请求堆栈管理策略

质量敏感型空间信息服务引擎服务请求堆栈管理的基本要求是:优先权要求高的空间信息服务要优先得到保障,相同优先权的服务请求,服务质量要求高的服务请求类型要优先得到保障。

服务请求堆栈管理策略应用在请求入栈和出栈 2 个阶段。在入栈阶段不同优先级要求的请求,进入不同的服务请求堆栈中,在高负载的情况下,对服务请求优先级较低的服务请求可被拒绝服务。

服务请求出栈的基本策略是优先级较高服务请求优先得到服务。但 2 种不同优先权的服务请求堆栈所采用的策略有所不同。对于优先级较高的服务请求,在一般负载下采用 FIFO(First In First Out)策略,在高负载下采用 LIFO(Last In First Out)策略,这样可以提高系统服务处理的吞吐量和减少服务请求等待的时间^[5]。不同负载的区分可根据当前堆栈中等待处理的服务请求数与系统预期处理能力的比例来区别。

在具体的每类堆栈中,服务请求出栈的基本策略是服务质量要求高的服务请求首先出栈。但为避免出现服务质量要求较低的服务请求长时间等待得不到处理,服务请求出列规则是如下权值来选取:

$$Pout=length \times priority$$

其中, $length$ 为某一类型服务请求队列中当前等待处理的服务请求的数量; $priority$ 是该类型服务请求服务质量要求的优先值。权值最大队列中的服务请求优先出栈得到服务。

4.2 服务请求分发策略

服务请求分发的任务,是把从服务请求堆栈中选取的服务请求派送到某一服务处理节点进行处理。请求分发策略是要保障服务请求被分发给最有可能在最短时间内完成该服务请求的服务处理节点。

服务请求分发在分布式系统负载均衡(Load Balancing)领域有非常多的研究,负载均衡策略基本上分 2 大类:(1)非适应型(Non-adaptive),如 Round-Robin 策略、Random 策略等;(2)适应型(Adaptive),这种类型策略能根据系统负载的不同调整行为,从而使请求总是被分发给负载最近或是较轻的服务器进行处理。

本文服务请求分发策略采用后一种,即适应型请求分发策略。在系统启动阶段,节点负载监测器首先获取每个节点的能力信息,即节点机器所包含的 CPU 数量、内存数量等,同时接收节点负载状态报告所发送的负载状态,分发策略根

(下转第 116 页)