

基于 Web Services 的异构空间信息共享

张 实, 肖卫东, 唐九阳
(国防科学技术大学 管理科学与工程系, 湖南 长沙 410073)

摘 要: 异构集成技术正日益成为信息资源管理的一个热点技术。以地理信息共享的应用为背景, 提出了基于 Web Services 的异构空间信息共享体系结构。通过引入 Web Services 技术架构, 设计了一个异构空间信息共享系统。该系统可提供各信息源灵活、动态发布自身服务的功能, 从而为快速构建异构集成系统提供了性能保证。
关键词: 异构空间信息共享; Web Services; WSDL; 统一描述、发现和集成; SOAP
中图法分类号: TP393 文献标识码: A 文章编号: 1001-3695(2004)10-0185-03

A Web Services-based Heterogeneous Geospatial Information Sharing

ZHANG Shi, XIAO Wei-dong, TANG Jiu-yang
(Dept. of Management Science & Engineering, National University of Defense Technology, Changsha Hunan 410073, China)

Abstract: Heterogeneous integration technology is increasingly becoming one of the hot focuses of information resources management. Based on the application of geospatial information sharing and Web Services technology, we propose and design a heterogeneous geospatial information sharing system. The system provides the function of publishing the sources' services flexibly and dynamically, hence making it possible to construct heterogeneous integration system immediately.
Key words: Heterogeneous Geospatial Information Sharing; Web Services; WSDL(Web Service Description Language); UDDI(Universal Description, Discovery and Integration); SOAP

空间信息技术涉及卫星通信、航天航空遥感、卫星定位和地理信息系统等专业领域,是快速获取大区域地球动态和定位信息的重要手段。随着计算机技术、网络技术及通信技术的迅速发展,通过网络实现空间信息共享势在必行。大量的空间信息源与高速互联网相连,为空间信息的获取提供了机遇,同时也带来了更大的挑战。由于参与共享的这些数据来自不同的部门、不同的组织,这些部门和企业使用的信息系统设备型号、信息存储方式、操作方式、通信方式、信息格式和执行标准以及信息可视化方式等都各不相同,产生了空间信息异构问题,使在异构环境下进行信息交流和共享产生了严重的问题^[1]。

目前,针对异构空间信息的特点,开发异构空间信息共享系统主要采用直接法和集成视图法^[2]: 直接法利用数据仓库、联邦数据库等将空间信息实现物理集成,形成具有统一模式的信息源,为用户提供空间信息,实现异构空间信息的共享。该方法的缺点是当异构信息源发生变化(如增添或删除)时,集成的信息源中的数据和模式要进行相应的修改,会出现信息更新不及时、重复存储等情况,信息源难以维护,所以实用性不高。集成视图法仍保持空间信息源的分布自治,仅增加了一个虚拟的集成视图以及这个视图与真实信息的映射关系;用户可以通过虚拟视图直接了解到需要的空间信息的存储位置、存储方式等情况,然后从信息源获取需要的空间信息。这种方法不需要重复存储大量数据,并能保证共享的是最新信息,适合高度自治且信息变化快的异构空间信息共享。但是当信息源发生频繁变化时,虚拟视图会变得难以维护。综合以上分析,

直接法和集成视图法都存在一定程度的不足。为了有效地利用异构空间信息和现存的处理系统,在最大范围内实现资源的共享和互操作,本文提出了基于 Web Services 的异构空间信息共享解决方案。

1 Web Services 架构

1.1 有关 Web Services 的概念

Web Services 是一套基于 XML 的标准,提供了一套全新的采用统一的方法进行服务的描述、注册、查询的模式,解决了信息孤岛问题,充分实现了信息的共享^[3]。也可以将 Web Services 理解为一种部署在 Web 上的对象(Web Object),以一种松散的服务捆绑集合形式动态地创建应用,是以 XML 为主的、开放的 Web 规范技术^[4]。

Web Services 的主要目标是通过使用统一标准,能够统一封装数据、消息、行为等,在无需考虑具体应用环境下让不同系统跨越平台,彼此兼容,进行无缝通信和数据共享。Web Services 实现的功能可以是响应一个简单的请求,也可以是完成一个复杂的商务流程。Web Services 体系结构基于三种角色: 服务提供者(Service Provider)、服务注册中心(Service Registry)、服务请求者(Service Requestor)之间的交互。交互具体涉及到服务发布(Publish)、查找(Find)和绑定(Bind)等三种行为^[5]。Web Services 体系架构模型如图 1 所示。

1.2 Web Services 主要协议^[6]

Web Services 主要包括三种协议,分别为 SOAP, WSDL, UDDI 协议。应用程序主要是通过 SOAP(简单对象访问协议)

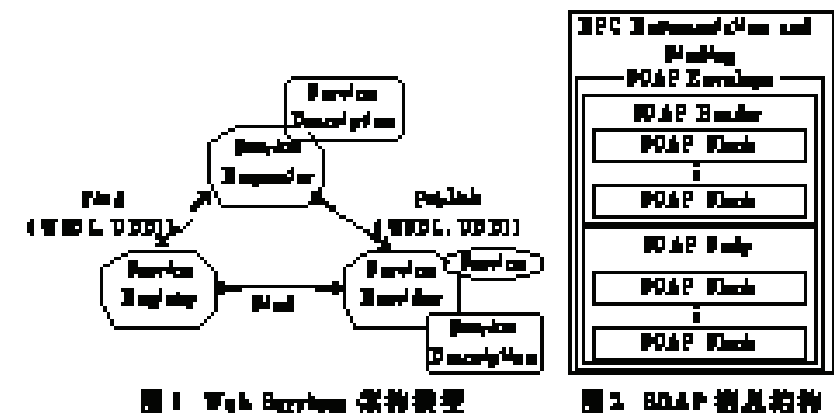
访问 Web Services; WSDL(Web Service Description Language) 是 Web Services 的描述语言, 用于描述与一个 Web Services 交互的消息格式、数据类型、操作、通信协议绑定、服务地址; Web Services 可以通过统一描述、发现和集成(Universal Description, Discovery and Integration, UDDI) 进行注册, 各信息源将信息源的信息以及它们所提供的 Web Services 的信息公开发布^[7]。下面对这三种协议分别进行介绍:

(1) WSDL 是一种基于 XML 的组件描述标准语言, 用于描述服务的内容、服务执行结果以及服务与通信协议进行绑定的信息, 为服务提供者描述服务提供了统一的标准, 同时用 WSDL 描述的服务信息也为用户了解服务、调用服务提供了条件。WSDL 提供一个抽象的语言, 利用参数和数据类型来定义被发布的操作、服务的位置和服务绑定的细节等。WSDL 不依赖于底层的协议和编码要求^[8]。

(2) UDDI 规范是基于分布式 Web 服务信息注册的规范, 为服务提供者提供一套标准的方法去注册登记相关信息, 并能去查询其他服务提供者所能提供的服务, 使用已经注册的服务, 分布、全球化地共享信息。它的主要功能是一种发布和发现 Web 服务信息的办法^[9]。

(3) SOAP 提供了一个简单、轻量的用于在分散或分布环境中交换结构化和类型信息的机制。SOAP 本身并没有定义任何应用程序语义(包括编程模型或特定语义的实现等)。它通过提供一个有标准组件的包模型和在模块中编码数据的机制, 定义了一个简单的表示应用程序语义的机制, 这使 SOAP 能够被用于从消息传递到 RPC 的各种系统中^[10]。

SOAP 包括三个部分: 封装结构、编码规则和 RPC 机制。其中, SOAP 封装结构定义了一个整体框架, 用来表示消息中包含的内容, 以及处理这些内容、属性的机制; SOAP 编码规则定义了用于交换应用程序数据类型实例的一系列机制; SOAP RPC 机制定义了一个用来表示远程过程调用和应答的协定。虽然这三个部分都作为 SOAP 的一部分一起描述, 但它们在功能上是相交的^[11]。SOAP 消息结构如图 2 所示。



2 异构空间信息共享系统的设计与实现

2.1 基于 Web Services 的异构空间信息共享框架(图 3)

根据前面介绍的 Web Services 技术, 并结合异构空间信息的特点以及当前异构空间信息共享技术的发展状况, 提出了基于 Web Services 的异构空间信息共享方案。

(1) 数据层。这部分是系统的最底层, 同时也是整个方案的基础, 主要包括数据源和适配器。数据源通过适配器将自身

所能实现的功能根据 WSDL 标准描述、封装为标准服务, 并将服务注册到空间信息 UDDI 注册中心; 当系统调用服务时, 由适配器负责激活服务, 实现相应的功能, 并将结果返回。

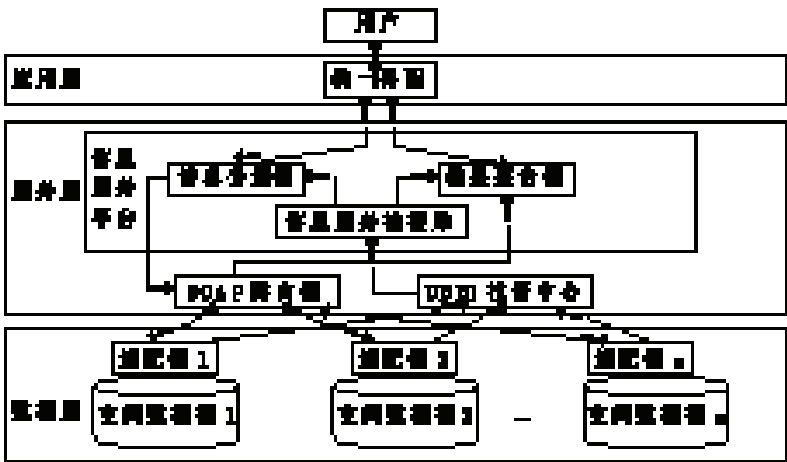


图 3 基于 Web Services 的异构空间信息共享系统框架

(2) 服务层。这部分是方案的核心, 主要包括 UDDI 注册中心、SOAP 路由器、信息服务台。信息服务台又包括了信息服务模型库、请求分解器、结果整合器。UDDI 注册中心记录与数据源所能提供的各种服务相关的信息, 并为信息服务台和 SOAP 路由器提供与服务相关的详细资料; 信息服务台负责将用户需要的共享信息请求分解为一系列针对具体信息源的子请求, 然后由 SOAP 路由器负责根据子请求调用具体的服务; 调用服务后获得的信息由信息服务台进行整合, 整合后的结果传递给统一用户界面。

(3) 应用层。这部分是方案的最顶层, 负责与用户的交互, 主要包括统一界面。通过统一界面, 用户可以在不需要知道具体实现细节的情况下提出共享请求, 经过服务层、数据层处理后, 获得经过整理的异构空间信息, 并最终展示给用户。

2.2 主要模块功能描述

该集成框架基于 Web Services 技术, 将数据源的提供信息的应用按照标准封装成 Web Services 服务, 然后将服务发布到 UDDI 注册中心; 在 UDDI 注册中心中可以获取服务的细节信息, 并能通过标准接口调用服务, 获取需要的空间信息。将提供信息的应用过程封装为标准的 Web Services 服务, 采用松散的耦合方式, 获取任何信息都可以通过激活对应的服务来实现, 服务可以根据要求动态地增添或删除, 因此这种方法实现简便, 使用灵活, 适于实现异构空间信息的共享。

空间信息源需要共享信息时, 首先将其能够提供的操作封装为服务, 并用 WSDL 对服务进行描述, 然后将服务用 SOAP 消息发布到 UDDI 注册中心。UDDI 注册中心根据使用范围不同, 分为私有注册中心和公有注册中心, 两种注册中心都可以支持异构空间信息的共享。

调用服务会激活信息源中的一组操作, 具体的操作由信息源制定。服务接口不发生变化, 服务的具体实现方法的变化不会影响到服务的调用, 也不会对信息服务模型产生影响, 真正做到表现与实现相分离^[12]。

(1) 适配器。它的主要功能可以分为两部分^[13]: 描述并注册服务。在这个功能中, 适配器将数据源中能够提供的操作和功能按照标准封装为服务, 然后调用标准的 UDDI API 将服务注册到 UDDI 注册中心。当需要服务时自动执行服务, 并

将服务执行结果返回。

(2) SOAP 路由器。它是实现 Web Services 的关键部件, 主要功能可以分为两部分: 接收由信息服务平台发送来的 SOAP 调用请求, 根据请求中的信息找出某个 Web Services 适配器的位置, 将请求消息传递给指定的适配器, 由适配器激活服务, 实现客户调用。接收由适配器传送的服务执行结果, 并将打包为 SOAP 消息形式的结果返回到信息服务平台。SOAP 路由器是实现 Web Services 的关键部件, 是服务调用和 SOAP 消息的传递核心。

(3) 信息服务平台。它主要负责信息共享请求的分解和请求结果的整合^[14]。其中信息共享模型库中存储了大量的信息共享模型, 它的主要作用是提供与服务相关的信息给请求分解器和结果整合器。请求分解器根据 UDDI 注册中心提供的服务信息和信息共享模型库提供的共享模型将用户的复杂的共享信息请求进行分解, 产生与具体信息源相联系的简单子请求, 简单子请求与独立的数据源上所提供的服务相对应; 由 SOAP 路由器负责将子请求发送给具体的适配器, 激活信息源上的具体服务, 服务执行后的结果返回给信息服务平台中的结果整合器; 结果整合器根据信息共享模型将各个信息源返回的具体信息进行整理, 并将整理结果传递给统一界面, 最终显示给用户。

(4) 统一界面。用户可以通过统一界面了解到自己从其他异构空间信息源所能获得的信息种类, 并根据需要提出请求, 统一界面会将请求打包生成 XML 消息, 并将消息传递给信息服务平台。这是统一界面的第一个主要功能。当请求结果由信息服务平台返回时也被打包为 XML 消息, 统一界面会将结果格式化显示给用户, 并且显示的形式可以根据用户的需求进行改变。

2.3 用户获取异构空间信息的工作流程(图 4)

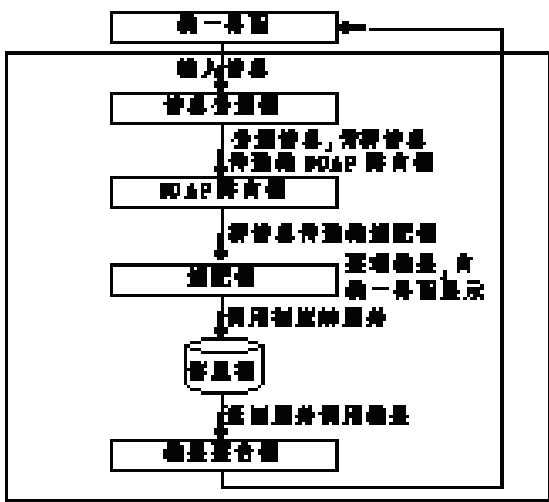


图 4 获取异构空间信息流程图

- (1) 用户通过统一界面提出请求, 请求分发给信息服务平台。
- (2) 信息服务平台中请求分解器得到请求后会与已有的信息共享模型进行匹配; 匹配成功后会根据信息共享模型将信息获取请求分解为一系列子请求, 这些子请求与独立数据源提供的服务对应。请求分解器将各个服务请求传递给 SOAP 路由器。
- (3) SAOP 路由器接收到服务请求后, 根据 UDDI 注册中心中的信息将服务请求递交给具体的适配器。
- (4) 适配器接收到服务请求, 激活服务, 获取执行后的结

果, 将结果返回给 SOAP 服务器。

(5) SOAP 服务器中结果整合器将结果打包为 SOAP 消息, 传回信息服务平台。

(6) 信息服务平台根据信息服务模型将从不同数据源获得的结果进行汇总, 并将汇总结果传递给统一界面。

(7) 统一界面负责将最终结果展示给用户。

3 结论

随着空间信息的不断增加, 用户迫切要求对异构空间信息进行集成。本文在分析了现有空间信息共享机制的基础上, 应用 Web Services 技术以及相关标准, 提出了基于 Web Services 的异构空间信息共享的基本框架。与传统的方法相比, 本文提出的方法具有用户使用信息不受地理位置的限制, 不受用户所使用的软件系统的限制, 没有明显时间上的滞后等优点; 同时这种方法避免了数据的重复生产, 避免了大量冗余数据的产生, 增强了对以前产生的数据的再利用, 并且在实际应用中获得了良好的效果。

参考文献:

[1] 易宝林, 曹忠升, 肖力. 异构的多媒体与空间信息共享[EB/OL]. <http://www2.ccw.com.cn/2000/0040/0040c08.asp>, 2000-10-16.

[2] 吴扬扬, 陈锻生. 一个基于 XML 的 Web 信息源集成方案[J]. 计算机工程与应用, 2001, (10): 34.

[3] 王莉莉, 王力生. Web Services 技术下的企业应用集成[J]. 计算机与现代化, 2003, (4): 39-40.

[4] 庄秀丽, 孙波. 基于 Web Service 异构教育资源库数据共享方案的研究与实现[J]. 电化教育研究, 2003, (2): 48-49.

[5] 吕曦, 王化文. Web Services 的架构与协议[J]. 计算机应用, 2002, 22(12): 62-63.

[6] B Lublinsky, M Farrell. Web Services: The Implementation Iceberg [J]. eAI Journal, 2002, (6).

[7] 刘刚, 余晖. 利用 WSDL 和 UDDI 为公共 Web Service 建立统一接口[J]. 计算机应用研究, 2003, 20(5): 150-152.

[8] Web Service Description Language v1. 1[EB/OL]. <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>, 2001-03-15.

[9] UDDI Technical White Paper[EB/OL]. <http://www.uddi.org/>, 2001-05-14.

[10] Simple Object Access Protocol v1. 1[EB/OL]. <http://www.w3c.org/TR/2000/NOTE-SOAP-20000508>, 2000-05-08.

[11] 程炜, 杨宗凯, 乐春晖. 基于 Web Services 的一种分布式体系结构[J]. 计算机应用研究, 2003, 20(3): 105-107, 111.

[12] 柴晓路, 梁宇奇. Web Services 技术、架构和应用[M]. 北京: 电子工业出版社, 2003. 347-349.

[13] K Hammer. Web Services and Enterprise Integration[J]. eAI Journal, 2001, (11): 69.

[14] Serge Abiteboul, Omar Benjelloun, Tova Milo. Web Services and Data Integration[C]. WISE, 2002. 3-6.

作者简介:

张实, 硕士研究生, 主要研究领域为分布式信息资源管理、智能决策支持技术; 肖卫东, 副教授, 博士研究生, 主要研究领域为信息资源管理、智能决策支持技术; 唐九阳, 博士研究生, 主要研究领域为信息系统集成、智能决策支持技术。