

基于 Active XML 的动态 Web 服务组合实现

靳 荣, 赵军富, 高永兵, 史海军

(内蒙古科技大学信息工程学院, 包头 014010)

摘要: 随着 Web 服务应用的发展, 大量存在的应用服务之间需要解决共享和交互, 动态 Web 服务组合为解决信息共享和应用协作的瓶颈问题提供了新的方案。该文通过引入 Active XML (AXML), 扩展 Web 服务, 提出了一个基于 Active XML 的动态 Web 服务组合框架, 在此基础上实现了一个动态服务体系结构。该体系结构支持组合服务的即时生成以及在动态环境中的快速部署和发布。

关键词: AXML; Web 服务; 动态 Web 服务组合

Dynamic Web Services Composition Based on Active XML

JIN Rong, ZHAO Jun-fu, GAO Yong-bing, SHI Hai-jun

(Information Engineering College, Inner Mongolia Technology University, Baotou 014010)

【Abstract】 With the development of the Web services application, it is necessary to resolve sharing and interaction in massive application services. Dynamic composition of Web services provides new solution to solve bottleneck brought by information sharing and application corporation. Through introducing active XML (AXML) and expanding Web services, this paper proposes dynamic composition Web services architecture based on AXML. The architecture supports dynamic services composition building in time, deployment and delivering in dynamic environment is implemented.

【Key words】 active XML; Web services; dynamic Web services composition

随着 Internet 技术的发展, Web 服务作为一种崭新的分布式计算模型, 实现了真正意义上的平台独立和语言独立。Web 服务被定义为使用标准的 XML 技术与其他服务进行交互的软件模块和自描述的应用。越来越多的企业将自己的业务作为 Web 服务发布, 企业对服务的速度、服务范围、服务质量等都提出了更高的要求。面对越来越多的 Web 服务, 如何快速部署与发现服务, 同时根据需把相关 Web 服务动态组合^[1]并发布为一个可以完成特定任务的新的服务, 以提高服务的即时性、减少服务的开发成本, 成为需要解决的关键问题。

在 Web 服务体系架构的各模块间, 消息是以 XML 格式传递。但是, 在数据交换过程中有些数据需要实时更新, 这样现有的 XML 无法做到及时的响应; 在这种背景下, INRIA (法国自动化研究所) 提出了一种分布式数据管理语言——Active XML^[2], 其特点是在 XML 中嵌入 Web 服务调用实现访问一些不断更新的数据, 能够通过调用 Web 服务对 XML 文档中某些数据实时给予准确的描述。

Web 服务组合已经在相关研究领域获得重视, 相关技术^[3]如: WISE, e-flow。它们大多需要处理底层的编程细节, 设计者需要考虑操作交互、消息之间的映射、服务调用的方法等。而基于 Active XML 的动态 Web 服务组合是利用 AXML 的动态特性对 Web 服务进行扩展, 对相关服务进行概念和定义的统一, 建立服务之间的关联, 达到灵活组合并提高组合质量。本文在 Web 服务描述语言 (WSDL) 的基础上引入 AXML, 实现 Web 服务适应具体环境的动态组合。

1 Web 服务组合分析

在异构系统中运行在不同平台上的 Web 服务可能是由不同供应商提供, 由不同的编程语言实现, 为满足服务的请求需要根据特定的应用背景和需求进行合理的 Web 服务组

合。Web 服务的组合潜在地缩短了开发时间, 减少了开发新应用的工作量。服务组合可分为静态组合和动态组合。静态组合是在设计时或装载时进行的, 它对一些变化不大的操作构成的系统是有效的, 但缺乏定制灵活性。动态组合是在运行时进行组合的, 它能给系统带来更大的灵活性。动态 Web 服务组合需要解决以下几个主要问题^[1]: (1) 怎样能够对基本 Web 服务中相关数据进行实时更新, 准确描述 Web 服务; (2) 怎样实现基本服务之间的协调、动态交互操作; (3) 怎样保证组合 Web 服务的有序执行。

2 动态 Web 服务组合主要技术

针对以上的几个问题本文引入 AXML^[4] (即动态 XML), 这种分布式数据管理语言, 对 Web 服务中相关数据实时更新。通过给定 Web 服务可组合性条件对 Web 服务进行可组合判断, 保证基本服务之间的协调、动态交互。提出一种数据结构解决组合 Web 服务有序执行问题。

2.1 AXML 对 Web 服务中相关数据实时更新

WSDL 是 Web 服务的描述语言, WSDL 通过定义一套 XML 的语法来描述 Web 服务。然而, WSDL 仅从语法格式上对 Web 服务进行表达, 本文引入 AXML 对 WSDL 扩展, 利用 AXML 的动态特性^[2], 对 Web 服务相关数据实时更新。下面是一个用 Active XML 对 WSDL 扩展 (见图 1), 进而实现 Web 服务实时更新简单示例:

```
<service name="weather forecast information consultation"/>
```

基金项目: 内蒙古自然科学基金资助项目 (200608010808)

作者简介: 靳 荣 (1960 -), 男, 硕士、教授, 主研方向: 计算机网络与 Web Services; 赵军富, 硕士研究生; 高永兵, 硕士、讲师; 史海军, 硕士研究生

收稿日期: 2006-09-25 **E-mail:** junfu_zhao@yahoo.com.cn

```

<purpose function=" weather forecast">
...
<binding name ="SOAP"/>
<message name ="query">
<parameter name=" date" type="string" .... />
<parameter name=" region" type="string" .... />
<parameter name=" temperature" type="float" unit=" degree" />
</message>
<operation name="receiveSpecialquery" mode="request-response" />
<input name=" query" />
...
</operation/>
<flowsource="..\..\newspaper.xml" target="replyqueryresult " >

```

图1 扩展的 WSDL 文档

在图1扩展 WSDL 文档中,顶层元素<service>,包括了服务的名字为 weather forecast information consultation 的属性。在<purpose>中指明此服务的功能 weather forecast, <binding>元素标签中指明绑定类型为 SOAP 类型,在<message>标签中指明消息的名字,同时列出消息的3个参数及每个参数的名字、类型等信息。提供所要查询的某日、某地区的气温等参数信息为输入信息,利用<flow>元素标签描述服务操作的流程控制,首先在整个天气预报服务的 WSDL 中通过指定图2这个 message.xml 文档的 URL 地址来执行该 AXML 文档,从而动态获得某地区的气温,然后由<replyqueryresult>提供咨询服务结果。

```

<?xml version="1.0" encoding="ISO-8859-1">
<newspaper xmlns="http://lemonde.fr"
xmlns:rss="http://purl.org/rss"
xmlns:axml="http://purl.org/net/axml">
<title>china daily</title>
<date>2006-04-16</date>
<weather>
<axml:call service="forecast@weather.com">
<city>Beijing</city>
</axml:call>
</weather>
</newspaper>

```

图2 一个 AXML 文档

图2是一个名为 message.xml 的 AXML 文档,通过 call service 调用 forecast@weather.com 来获取 2006 年 4 月 16 日北京气温。

2.2 Web 服务的可组合性判定

在获得了 Web 服务实时更新后,对服务的可组合性进行判断,确定一个服务是否为可组合,来实现 Web 服务动态组合。本文提出了 4 级的可组合性判定^[5],保证组合的正确性。

(1)mode 可组合性。为使两个服务能够进行可组合交互操作,在客户/服务两端的操作一定有相一致的模型,根据现有的调用形式定义下面 4 种调用操作模式:1)notification:具有单向性,发起请求消息但无须接收任何回应消息。2)receive:操作接收一个输入消息并处理它,但不产生任何输出消息。3)invoke-response:操作生成一个输出消息,再接收一个相应的输入消息。4)request-response:请求应答式,操作接收一个消息,处理它并产生相应的应答输出。

(2)binding 可组合性。保证了操作在模式上的可组合后,还必须保证由于不同的服务可能支持不同的 binding 协议(如 SOAP, HTTP),这些服务能够彼此理解通信协议,以确保正常交互。

(3)操作可组合性。确保要交互的操作之间有相关的目的和分类。如果两个操作提供不同的功能,它们之间将不能做出映射。

(4)消息映射一致性。一个消息包含多个参数,每个参数都有相应的数据类型。Web 服务消息交互需要检查被传递来的参数的数据类型是否与接收者所需参数的类型一致,确保传递参数的数据类型包含于接收参数的数据类型。

2.3 组合 Web 服务的执行顺序

对于组合 Web 服务来说会同时有多个 Web 服务实例运行,因此,要保证当一个消息到来时,能够被传递到组合 Web 服务的正确实例。下面用一个数据结构来解决这个问题:

服务编号	服务名	操作名	操作类型	下一操作节点	入口地址	操作类型
------	-----	-----	------	--------	------	------

每一个操作都保存有下一个操作的入口信息。服务组合执行引擎会解析扩展的 WSDL 文档,确定服务的执行顺序并做出状态记录。当信息到来时,启动相应操作,并记录相关操作信息。执行完成的操作结果将作为下一操作的输入被传递。同时,更新执行记录的当前数据和执行状态。如执行中出现等待异步消息,此时服务会等待,直到消息到来相应的操作才能被激活。如果服务在等待过程中被终止,则判断结果是否可用,如果可用,提交并永久存储入数据库。

3 动态 Web 服务组合的体系结构与实现

3.1 动态 Web 服务组合的体系结构

动态 Web 服务组合的体系结构^[6]如图 3 所示,主要划分为 4 层。底层为数据存储层,包含有不同的信息库。每一个信息库对应一个可以对它进行访问的服务组件。在组合服务层,服务组件就是关于每一个单个数据源的封装器。这个封装器的作用就是抽取和打包查询结果到一个 AXML 文档,最后将该文档返回给上层调用。它提供了一个统一的访问接口以使用 WSDL 语言把相关类别的服务组件描述为一类服务,通过一定的组合判定规则进行服务组合生成新的服务。推理层主要将应用表示层传来的服务调用在推理逻辑中进行处理,同时会解析请求的合理性,并通过规则对服务请求进行标准化和约束设定。经过推理逻辑的处理后,用户的请求将被重构为对信息源的请求。

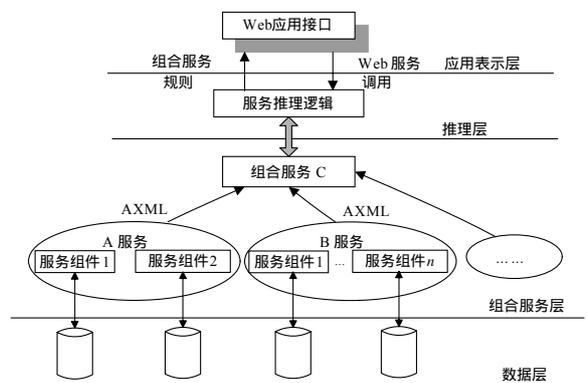


图3 动态 Web 服务组合的体系结构

3.2 动态 Web 服务组合过程及其组合实现

动态 Web 服务组合合成模块^[7]是动态 Web 服务组合结构的核心,由于篇幅所限本文仅对合成模块详细介绍,对其他部分不作介绍。动态 Web 服务组合结构如图 4 所示。

合成模块主要包含 4 个子模块:(1)需求预处理:接受用户输入的查询条件,并对其进行概念的标准化。(2)服务查找器:利用服务属性等描述信息查找符合要求的多个服务,并解析服务组件的 WSDL 文档,将服务的绑定类型、消息类型、参数信息、操作信息等提取存放为下一步服务组件的可组合性判定所用。(3)服务可组合性判断:服务可组合性判定器通过 2.2 节提到的可组合规则对已查找到的服务进行可组合判

断。(4)组合服务描述包装器:将可组合的服务生成新的描述。该描述文件有一系列的<Link>元素,每一个<Link>元素将组合服务的一个操作与相应的组件服务的操作相连接。并且声明为了完成一个活动这两个操作之间所要完成的交互。新生成的描述文件将被重新注册到服务注册容器 UDDI 中。

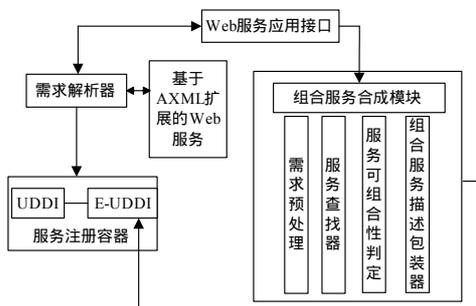


图 4 动态 Web 服务组合结构

合成模块首先根据客户端提出的需求,查找已注册服务组件,并把它们插接在一起,形成组合服务。然后,这个组合服务会被重新注册到 UDDI 中供用户调用。当 Web 应用接口接收服务请求后,将服务请求对照相应动态更新了 Web 服务进行概念统一标准化,标准化后的服务请求被作为输入条件传入下一级的服务查找器,服务查找器将列出所有符合条件的服务的描述,并调用服务可组合性判定模块对这些服务进行组合判定,将再次筛选出的服务的描述提供给组合服务的描述包装器模块,由包装器对各个服务描述进行组合,形成相关组合服务的描述,提交给专用的 UDDI 注册中心进行注册。

4 结论与展望

(上接第 111 页)

如果配置更多的接口机,虽然不会减小延迟,但会显著提高整个服务器对 IPv6 数据包吞吐量。

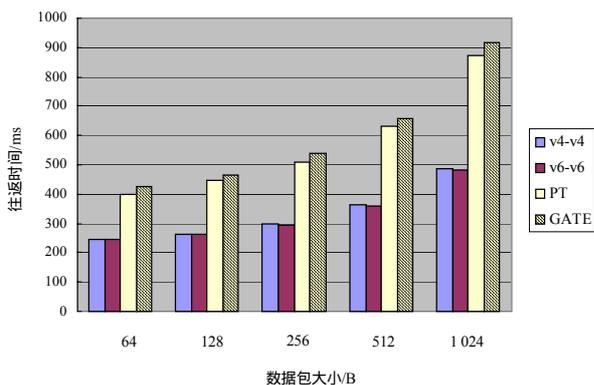


图 7 PING 数据包往返延迟

4 总结

本文提出了一种 NVS 的 IPv4/IPv6 接入机制,并通过将协议转换内置到接口机的内核中来实现。该机制能使多接口机的系统软件和上层应用软件透明使用,其延迟小于通过 NAT-PT 网关实现的延迟,同时具有更高的吞吐量。

随着 IPv6 的不断推广,未来工作主要是解决如下问题:将 IPv4/IPv6 协议转换软件移植到商用接口机中;在安全上,

在对 Web 服务深入应用的趋势下,为了能够依据业务需求,快速、灵活地集成各种已有的 Web 服务,本文提出基于 AXML 的动态 Web 服务组合,给出组合的体系结构,同时对 Web 服务组合的执行顺序、合成模块、组合的实现作了详细的介绍。对于动态组合服务的有效性验证及安全等问题,有待进一步的研究和探索。

参考文献

- 1 岳 昆, 王晓玲. Web 服务核心支撑技术: 研究综述[J]. 软件学报, 2004, 15(3): 428-442.
- 2 Abiteboul S, Benjelloun O, Milo T, et al. Active XML: Peer to Peer Data and Web Services Integration(demo)[C]//Proc. of VLDB'02. 2002.
- 3 Casati F, Ilnicki S, Jin L, et al. Adaptive and Dynamic Service Composition in eFlow[C]//Proc. of the CASISE Conference, Stockholm. 2000: 13-31.
- 4 靳 荣, 史海军, 高永兵, 等. AXML 在基于 Web Services 的工作流过程建模中的应用[J]. 微计算机信息, 2006, 22(6): 253-255.
- 5 Medjahed B, Bouettaya A, Elmagarmid A K. Composing Web Services on the Semantic Web[Z]. (2003-05). http://www.cs.purdue.edu/homes/ake/pub/composition_VLDBJ.pdf.
- 6 Thakkar S, Knoblock C A, Ambite J L, et al. Dynamically Composing Web Services from On-line Sources[Z]. (2002-09). <http://www.aaai.org>.
- 7 Benatallah B, Dumas M. Definition and Execution of Composite Web Service: The SELF-SERV Project[J]. Bulletin of the IEEE Computer Society Committee on Data Engineering, 2002, 25(4): 47-52.

完善与标准的安全协议集 IPsec 的协同运作。

参考文献

- 1 Lee Joo-Chul, Shin Myung-Ki, Kim Hyong-Jun. Implementing NAT-PT/SIIT, ALGs and Consideration to the Mobility Support in NAT-PT Environment[C]//Proceedings of the 6th International Conference on Advanced Communication Technology. 2004: 433-439.
- 2 Chen Whai-En, Su Chia-Yung, Weng Jui-Hung. Development of IPv6-IPv4 Translation Mechanisms for SIP-based VoIP Applications[C]//Proceedings of the 19th International Conference on Advanced Information Networking and Applications. 2005: 819-823.
- 3 Zhao Xiaoyu, Ma Yan. Linux Based NAT-PT Gateway Implementation[C]//Proceedings of 2001 International Conferences on Info-tech and Info-net. 2001: 258-263.
- 4 侯宗浩, 郑守淇, 董小社, 等. 多入口单 IP 集群接口方案及其实验分析[J]. 小型微型计算机系统, 2004, 25(6): 961-964.
- 5 伍卫国, 董小社, 刘爱华, 等. 集群服务器系统前端接口子系统的设计与实现[J]. 西安交通大学学报, 2005, 39(8): 787-791.
- 6 杨晓宁, 伍卫国, 刘爱华, 等. 多负载均衡器集群系统中负载均衡器故障恢复机制[J]. 计算机工程, 2004, 30(12): 45-46.
- 7 Hong Yong-Geun, Shin Myung-Ki, Kim Hyoung-Jun. Application Translation for IPv6 at NAT-PT[C]//Proceedings of the 9th Asia-Pacific Conference on Communications. 2003: 203-207.