

# Web 服务的绑定与调用方法研究

李 苏, 刘建勋

(湖南科技大学 知识处理与网络化制造湖南省普通高校重点实验室, 湖南 湘潭 411201)

**摘 要:** Web 服务是一个崭新的分布式计算模型, 是 Web 上数据和信息集成的有效机制。然而, 随着 Web 服务的日益增多, 传统静态的 Web 服务调用方式已不能满足人们的需求, Web 服务调用的静态或半自动化一直是面向服务应用的瓶颈所在。从不同侧面综述了国内外在该领域的最新研究现状, 分析了服务调用的方式和难点, 比较了静态调用和动态调用的区别, 给出了服务动态绑定与调用的基本方法和步骤。从而降低服务双方的耦合性, 使 Web 服务在分布式计算环境下发挥更大的灵活性。

**关键词:** Web 服务; 动态绑定与调用; 接口匹配

**中图分类号:** TP301

**文献标识码:** A

**文章编号:** 1673-629X(2010)06-0059-04

## Research on Methods of Web Service Binding and Invocation

LI Su, LIU Jian-xun

(Key Laboratory of Knowledge Processing and Networked Manufacturing, College of Hunan Province, Hunan University of Science and Technology, Xiangtan 411201, China)

**Abstract:** Web service is a brand-new distributed computing model, and it is an effective mechanism to integrate the data and information on the Web. However, with the increasing numbers of Web services, traditional static Web service invocation cannot satisfy people's needs any more, semi-automatic or automatic invocation is always the bottleneck of Web service invocation in service oriented applications. Summarize the research on Web service binding and invocation from different aspects at first. It then analyzes the manner of service invocation and difficult, compares between the static and dynamic invocations, gives the basic steps of service dynamic binding and invocations, thereby reducing the coupling of both services side, and makes Web services play a greater flexibility in a distributed computing environment.

**Key words:** Web services; dynamic binding and invocation; interface matching

## 0 引言

随着 Web 服务技术的发展, 面向服务计算逐渐成为开放异构环境中复杂分布应用的主流计算模型, Web 服务组合作为一种将多个自治的 Web 服务按照一定逻辑顺序组合成功能更强大的 Web 服务方式, 为复杂的 Web 应用提供了有效的解决方案<sup>[1~4]</sup>。为了充分发挥 Web 服务的灵活性, 用户必须能够动态地发现和调用 Web 服务。在服务调用过程中, 数据在服务提供者和服务消费者之间进行传递, 由于 Web 服务由不同的服务提供者开发, 而每一组织可能有它自己的

商务规则, 因此 Web 服务消息参数的定义往往建立在不同的领域中, 消息结构也可能存在差异, 而且由于用户在请求时不可能知道存在什么样的服务, 服务提供者在提供服务时也不能预期都会存在什么样的请求, 服务消费者对请求的 Web 服务无法自动的“理解”它传递过来的消息, 由此导致用户的服务请求和服务提供者发布的服务的完全匹配只是一种最理想状态, 在实际进行匹配时绝大部分不是完全匹配<sup>[5,6]</sup>。从而导致了 Web 服务不能直接应用于组合业务流程, 限制了业务合成的自动性, 降低了 Web 服务发现的准确性、有效性及 Web 服务的使用效率。因此如何消除服务请求描述与服务提供者提供的服务之间消息的异构性并实现 Web 服务的动态绑定与调用, 已成为 Web 服务应用领域中一个亟待解决的问题。

收稿日期: 2009-10-18; 修回日期: 2010-01-22

基金项目: 国家自然科学基金(90818004); 湖南省科技计划项目(2007GK3054)

作者简介: 李 苏(1985-), 男, 硕士研究生, 研究方向为 Web 服务动态绑定与调用、可信 Web 服务组合, 服务计算; 刘建勋, 博士, 教授, 主要研究方向为语义与知识网络、工作流、Web 服务等。

## 1 Web 服务

文献[7]描述了 Web 服务的基本体系结构, Web

服务体系结构由三种角色和三种基本操作构成。三种角色包括:服务使用者、服务提供者和服务注册中心。Web 服务体系结构中的每个实体都扮演着服务提供者、使用者和注册中心这三种角色中的某一种(或多种)。同时,Web 服务体系结构中还包括三种操作:

\* 发布。为了使服务可访问,需要发布服务描述以使服务使用者发现和调用它。

\* 发现。服务请求者定位服务,查询服务注册中心来找到满足其需求的服务。

\* 绑定和调用。检索完服务描述之后,服务使用者根据服务描述信息来调用该服务。

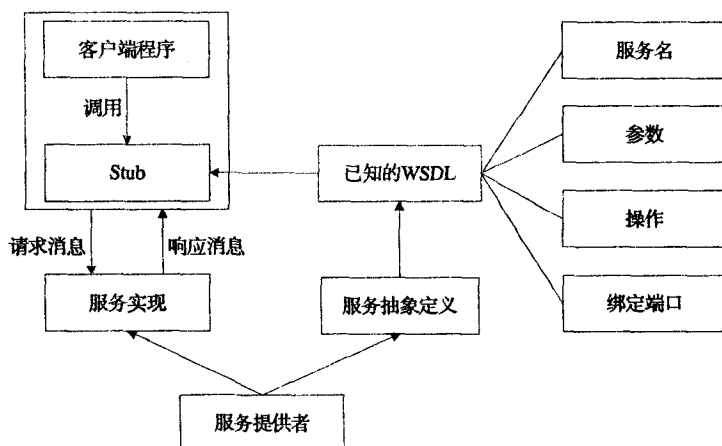


图 1 服务的静态调用

## 2 Web 服务绑定与调用方式和问题

Web 服务允许不同种类的应用程序通过标准 HTTP 协议调用已发布的方法。过去的大部分远程调用技术(例如 Java RMI 或 CORBA)都在二进制层操作并且要求紧密耦合<sup>[8,9]</sup>。而 Web 服务通过使用 XML、SOAP 和 HTTP 等组件,使得服务使用者和服务提供者在服务实现和客户如何使用服务方面隔离开来,服务接口作为与服务实现分离的实体而存在,使用者无须了解提供者的实现细节,因此 Web 服务是松散耦合的,其数据交换以 XML 格式通过 HTTP 进行,并以静态、半自动或者动态方式调用 Web 服务。

### 2.1 静态绑定与调用

服务的静态绑定与调用是指服务使用者在调用服务之前就已经掌握了服务的功能、接口参数、绑定类型与方式等信息,通过静态输入消息参数值进行调用的过程,在这个过程中,服务对于请求者来说是可理解的<sup>[10]</sup>。如图 1 所示,服务通过 WSDL 进行抽象描述,在服务调用之前,客户端程序员采用 WSDL2Java 工具预生成客户端 Stub。该 Stub 符合预先约定的 WSDL 规范,它是服务端在客户端的辅助对象,在服务调用时,客户对象先调用 Stub,再由 Stub 调用网络远端的服务辅助对象,而服务辅助对象最终调用真正的服务对象。静态调用意味着客户端程序员需要人工对 WSDL 进行分析、理解,了解所需要的服务的功能,特别是服务操作,参数及其类型,在调用服务时,通过封装这些消息参数,调用所需要的操作,发送请求到服务的实现端。这样,客户端将服务操作和参数固化在客户端程序中,因此,客户端和服务提供者之间紧密地耦合在一起,一旦服务提供方更改服务的某些细节,客户端不得不重复上面的工作来保证自己的程序不出错。

### 2.2 动态绑定与调用

动态绑定和调用的目标是对服务消费者与服务提供者之间进行解耦,尽量减少服务调用过程中的人为参与。对用户而言,不需要了解服务的实现过程及 WSDL 中参数的结构和意义,不需要担心服务的改变导致自己程序的崩溃。

一个服务可以被建模为一系列的前置和后置约束。因此,服务的功能匹配就可以通过检查一系列给定的初始约束。

定义 1: 一个 Web 服务可用下面的表达式描述:  $WS_i(Pre_i, Post_i)$ 。其中,  $WS_i$  是 Web 服务的名字,  $Pre_i$  是该服务的前置条件集合,  $Post_i$  是该服务的后置条件集合。

定义 2: 一个 Web 服务请求可以用下面的表达式描述:  $WSR_j(Pre_j, Post_j)$ 。其中,  $WSR_j$  是 Web 服务请求的名字,  $Pre_j$  是该服务的前置条件集合,  $Post_j$  是该服务的后置条件集合。

服务的动态调用如图 2 所示,当服务请求到来时,首先利用匹配算法构建相应的匹配器,然后输入用户的  $WSR_j(Pre_j, Post_j)$ ,并从 WSDL 集合中提取服务描述,通过匹配器将两个描述进行匹配。匹配时,需要 Ontology<sup>[10]</sup>本体库中的相关信息与接口映射规则来确定它们之间的关系,最后输出满足用户需求的服务。Ontology 本体库中存放了在本领域中所有服务的提供者达成共识的领域本体,其中包括:类和属性的分化与它们之间的关系、类的继承关系、类间的二元关系、定义良好的术语集、功能、输入事件签名以及在节点交互时涉及对象的逻辑语义。

### 2.3 Web 服务的调用步骤

一般而言,服务调用由以下几个基本步骤组成:

(1) 装载 WSDL 文档,选择满足条件的 WSDL 装载,如满足一定 QoS 的服务描述。

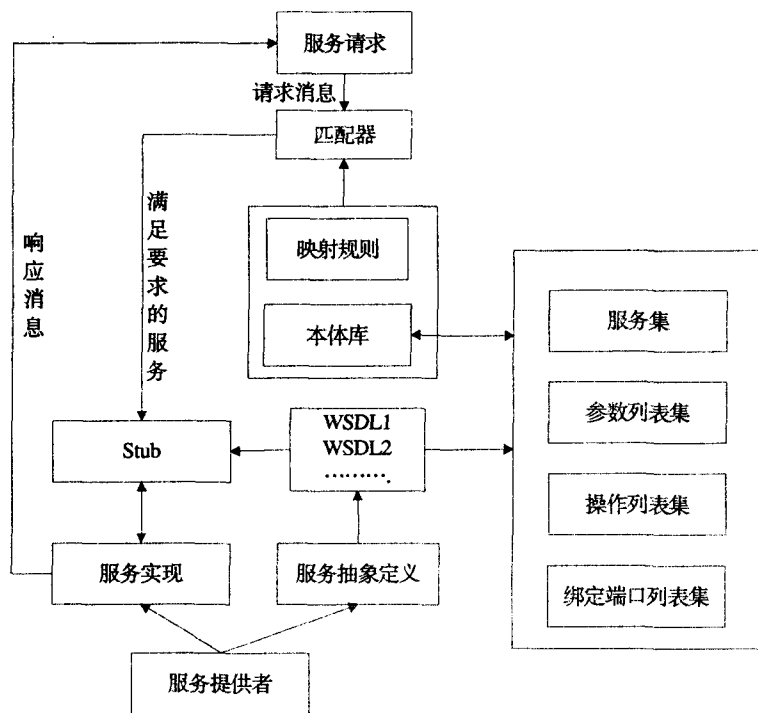


图2 服务的动态调用

(2)解析 WSDL 文档,分析其端口类型、参数类型及名称、服务操作、调用协议和机制等。

(3)根据用户的需求选择合适的适配器来绑定端口,选择绑定类型。

(4)根据输入输出参数创建输入消息体,对 WSDL 描述的服务参数和用户提供的服务参数进行映射。

(5)通过给选定端口提供待调用操作的名称以及操作需要的一个输入和(或)输出消息进行调用。

(6)返回并解析服务调用结果。对满足用户需求的多个服务调用的结果的优劣进行评判,选择最优的结果返回。如用户调用一个网上购书服务,当存在多个提供网上购书的服务时,用户需要返回某本书最低价格的服务调用结果。

## 2.4 绑定与调用的问题

### 2.4.1 Web 服务多样性带来的挑战

Web 服务自包含、自描述、模块化和松耦合等特点使 Web 服务不可避免地就会存在描述结构异构、语法异构和语义异构等问题。Web 服务的多样性给 Web 服务调用带来了挑战。

首先,描述结构异构是指用不同的数据结构和层次定义接口参数描述结构,使 Web 服务之间存在主要形态、运行和使用方式的差异。在 Web 服务中为了解决对形态万千的 Web 服务的访问和互操作问题,业界提出了 WSDL 协议,即采用变量类型(Types)、消息(Message)、端口类型(PortType)、绑定(Binding)和端口(Port)五种元素来描述服务接口的语法。

语法异构主要是因 Web 服务描述的不规范性,而存在描述的不完整性、不一致性、不确定性等问题。UDDI 等语法级 Web 服务描述语言试图利用关键字和预定义分类描述来集中地描述和存储 Web 服务,以解决语法异构问题。

语义异构是指使用不同术语来表达同一接口参数。为解决这两个异构问题,研究人员提出了基于本体论的语义描述规范,如 OWL-S、WSMO、SAWSDL<sup>[11]</sup>等。本体试图为 Web 服务的描述提供共享的、精确定义的语义术语。可是,目前对于本体的建立、更新,不同抽象级别的本体语义术语的精确定义,整个 Web 上的分布式本体的一致性维护等,这些问题会使语义描述规范依然存在一定程度的语法、语义异构问题。SAWSDL 为 WSDL 添加语义标注提供了一种机制,它从本体或映射文件中引用一个概念,将语义注释和语义模型相关联,不过目前关于 WSDL 的语义标注工作还处于研究阶段。

### 2.4.2 Web 服务调用的灵活性问题

Web 服务的调用效率和性能和下面一些客观因素紧密相关:被调用服务的请求连接数的多少、服务请求方和服务提供方的网络负载情况、服务的质量等等。为了解决以上问题,Aliaksandr Birukou 等人提出了建立用户反馈模型,通过反馈模型来指导后面的用户选择合适的服务;CMU 大学的 OWL-S 虚拟机<sup>[12]</sup>通过将服务调用计划与服务调用执行分开,实现服务的灵活调用。

### 2.4.3 Web 服务调用的安全性问题

随着 Internet 上 Web 服务的日益增多,越来越多的企业和个人利用 Web 服务来集成他们的业务,Web 服务在安全方面的实践和挑战也随之发展。Web 服务调用是通过客户端发送请求消息到服务提供方并获得请求响应的过程,在这过程中涉及到许多安全的问题,比如身份验证,服务授权,消息的完整性、机密性和不可否认性等等,这些安全问题对于像银行这类涉及个人信息隐私的服务尤为重要。针对 Web 服务的安全问题,W3C 和 OASIS 等组织发布了一系列标准,如 WS-Security, WS-Trust<sup>[13,14]</sup>等,越来越多的企业和组织也在不断开发和更新标准的实践。

## 3 Web 服务绑定与调用研究现状

### 3.1 调用框架

Web 服务的流行促使业界企业和组织开发构建

服务和调用服务的工具,其中代表性的有 WSIF, Axis, Xfire, Apache CXF 等。WSIF 是由 IBM 于 2001 年提出的独立于 SOAP 绑定的服务调用框架,它提供一套 API,使程序员无需考虑 Web 服务的传输协议及其具体位置,并且允许以相同的方式调用非 SOAP 的服务。Apache Axis 是一个基于 Java 语言的 Web Services 服务和客户端的实现,Apache Axis 提供了一个完整的对象模型和模块化体系结构,可以让你很容易地接收和处理 SOAP 消息,从一个普通的 Java 类建立 Web Service,发送和接收带有附件的 SOAP 消息,建立或者使用基于 REST 的 Web Service 等等。XFire 框架简化了 Java 应用转化为 Web 服务的步骤和过程,降低了 SOA 的实现难度,为企业转向 SOA 架构提供了一种简单可行的方式。Apache CXF 采用代码优先(Code First)或者 WSDL 优先(WSDL First)来轻松地实现 Web Services 的发布和使用,并为 Web 服务的安全性提供了更多的支持。

### 3.2 语义匹配

目前国内外研究主要是从概念匹配和结构匹配这个角度研究服务接口语义匹配问题。随着自然语言处理技术的发展,语义的研究,概念匹配主要是词汇语义的研究。李熙<sup>[15]</sup>等利用 WordNet 语义字典,分别从语义概念树的深度和宽度考虑它们对词语语义距离的影响,并结合权值,给出了一个计算词语语义相似度的公式;李涓子<sup>[16]</sup>利用这种思想来实现语义的自动排歧;吴芸<sup>[17]</sup>研究了如何利用词语的相关性来计算词语的相似度。结构匹配主要是 XML 消息模式匹配(Schema Match),于守健<sup>[18]</sup>等结合现有的 WordNet 语义词典,提出了服务接口匹配算法,计算接口之间的相似度;李涛<sup>[19]</sup>等提出了基于本体的消息匹配及重用机制,从 Web 服务的数据接口部分抽取本体概念,使用这些概念存储已有的匹配结果,并通过组合的方式产生新的匹配记过来提高匹配的成功率和召回率,辅助服务绑定的用户操作。

## 4 结束语

对于 Web 上千差万别的服务资源,如何建立一个合适的领域本体;如何保证较高的效率和灵活性的同时,机器能自动选择和调用合适的 Web 服务等问题还有待解决。特别是随着 Web 服务技术的应用,在 Web 服务部署方法、服务请求的精确理解、服务匹配算法等方面还有广阔的研究空间。目前,在语义 Web 服务、知识发现、机器学习等相关领域取得了一些突破性的研究成果,其中一些解决关键问题的方法和途径,将有助于 Web 服务动态调用方法的研究。因此,在当前

Web 服务绑定与调用的研究基础上,合理借鉴语义描述、结合已有的服务调用框架,机器学习等其他相关技术,将可能为 Web 服务动态绑定调用提供更加新颖和有效的方法。

### 参考文献:

- [1] Buhler P, Starr C, Schroder W H, et al. Preparing for Service - Oriented Computing: A Composite Design Pattern for Stubless Web Service Invocation [M]. Berlin, Heidelberg: Springer, 2004: 763 - 769.
- [2] Leitner P, Rosenberg F. DAIOS - Efficient Dynamic Web Service Invocation [J]. IEEE Intelligent Systems, 2009, 13 (3): 72 - 80.
- [3] Christensen K, Olesen T H, Thomsen L L. Matching Semantically Described Web Services Using Ontologies [J]. Information Technology and Control, 2006, 35(3a): 267 - 366.
- [4] ZHANG P. Travel Ontology [EB/OL]. 2005. <http://www.schemaWeb.info/schema/SchemaDetails.aspx?id=236>.
- [5] Akram M, Medjahed B, Bouguettaya A. Supporting Dynamic Changes in Web Service Environments [C]//ICSOC 2003. [s. l.]: [s. n.], 2003: 319 - 334.
- [6] Burstein M. Dynamic Invocation of Semantic Web Services That Use Unfamiliar Ontologies [J]. IEEE Intelligent Systems, 2004, 19(4): 67 - 73.
- [7] 廖祝华, 刘建勋, 刘毅志, 等. Web 服务发现技术研究综述 [J]. 情报学报, 2008, 27(2): 186 - 192.
- [8] Cavallaro L, Nitto E D. An Approach to Adapt Service Requests to Actual Service Interfaces [C]//SEAMS'08. Leipzig, Germany: ACM, 2008.
- [9] Ardagna D, Comuzzi M, Mussi E, et al. Paws: A framework for executing adaptive Web - service processes [J]. IEEE Software, 2007, 24(6): 39 - 46.
- [10] Nezhad H R M, Benatallah B, Martens A, et al. Semi - automated adaptation of service interactions [C]//WWW '07: Proc. of the 16th international conference on World Wide Web. New York, NY, USA: ACM Press, 2007: 993 - 1002.
- [11] Nezhad H R M, Saint - Paul R, Benatallah B, et al. Service-mosaic: Interactive analysis and manipulation of service conversations [C]//ICDE. [s. l.]: ACM Press, 2007: 1497 - 1498.
- [12] 潘长胜, 于浩海, 王光兴. Web Services 动态整合的体系结构和算法 [J]. 东北大学学报: 自然科学版, 2003, 24(5): 445 - 448.
- [13] Li X, Fan Y, Jiang F. A classification of service composition mismatches to support service mediation [C]//GCC '07: Proc. of the Sixth International Conference on Grid and Cooperative Computing (GCC2007). Washington, DC, USA: IEEE Computer Society, 2007: 315 - 321.

具体方面织入时只需要指定方面的具体信息,上述定义的规则等价于下面的表示:

1) `reinterpretEvent(core, "channel_Idle", "request", "initState", "mode = VHF || mode = UHF", AspectID, statechar. PREHANDLE);`

2) `reinterpretEvent(core, "Transmit_Transmit_Radio_VHF", "push", "initState", "Q1", Aspect - ID, statechar. PREHANDLE);`

3) `reinterpretEvent(core, "message_in_buffer", "output", "initstate", "Q2", AspectID, statechar. PREHANDLE);`

织入时每个状态图之间不必指定其他的状态图的具体细节,方面状态图可以应用于任何需要的地方。由于没有严格限制状态图之间的横切关系,一个方面状态图可以横切核心状态图或其他方面状态图。在这里,只简单地对 QoS 方面进行织入的分析,建立 QoS 的面向方面模型可以利用 CASE 工具实现代码自动生成。

### 3 结束语

文中通过引入面向方面技术,对网格建模,网格环境下对 QoS 要求很严格,而且 QoS 分散在设计中的各个方面,致使资源管理和调度很困难,而文中把 QoS 分离出来构造为一个单独的 QoS 方面,建立一个比较完整的 QoS 模型来建模系统服务质量,促使 QoS 方面与系统其他方面分开,能够进行单独的 QoS 方面设计。同时又能够根据需把设计好的 QoS 方面织入到系统中,从而简化了网格环境建模的复杂性。

文中只涉及到 QoS 方面建模和织入,但网格环境通常都非常复杂,涉及到许多方面,而且方面之间也存在交互横切关系,虽然文中提到的隐含编织方法有很好的扩展性,可以向系统加入新增或其他的方面,而不会影响已有的方面和主要功能,但还需要进一步的改进。这种设计方法,需要以下的完善:

- 1) 进一步扩展 UML 以能更好地建模 QoS 方面;
- 2) 进一步把传统 QoS 模型和面向方面的模型和面向方面的建模方法结合起来;
- 3) 进行与 CASE 工具结合,实现方面代码的自动产生与织入。

### 参考文献:

- [1] 梁 泉,杨 扬. 网络环境下的服务质量(QoS)研究[J]. 计算机科学,2006,32(7):42-46.
- [2] 李志纯,张南平. 面向 Aspect 编程的应用研究[J]. 计算机技术与发展,2006,16(5):217-218.
- [3] Alawud O, Bader A, Ellrad T. Weaving with Statecharts[C]// Aspect - Oriented Modeling with UML workshop at the 1st International Conference on Aspect - Oriented Software Development. Boston: [s. n.], 2002.
- [4] Booch G, Rumbaugh J, Jacobson I. UML 用户指南[M]. 北京:机械工业出版社,2000.
- [5] Rumbaugh J. UML 参考手册[M]. 姚淑珍,唐发根,等译. 北京:机械工业出版社,2001.
- [6] Alawud O, Bader A, Ellrad T. UML profile for Aspect - oriented software development[C]//In Workshop on Aspect - Oriented Modeling with UML (AOSD). Boston: [s. n.], 2003.
- [7] OMG. UML profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms Specification[EB/OL]. 2008. www.omg.org.
- [8] Cross J K, Schmidt D C. Applying the Quality Connector Pattern Optimize Distributed Real - time and Embedded Middle-ware and Applications[M]. London: [s. n.], 2002.
- [9] 何良玉,张立臣. 基于 UML 的网格 QoS 面向方面建模方法[J]. 现代计算机,2007(12):4-7.
- [10] 刘瑞成,张立臣. 基于 UML 的面向方面建模方法[J]. 计算机科学,2005,32(10):204-209.
- [11] Filman R, Elrad T, Clarke S. 面向方面的软件开发[M]. 北京:机械工业出版社,2006.
- [12] 刘瑞成,张立臣. 基于 UML 的面向方面的实时系统建模方法[J]. 计算机应用,2005,25(8):1874-1877.
- [13] 刘瑞成,张立臣. 基于 UML 的面向方面的实时系统建模方法[J]. 计算机应用,2005,25(8):1874-1877.
- [14] Penta M D, Esposito R, Villani M L, et al. Ws binder: a framework to enable dynamic binding of composite Web services[C]//SOSE '06: Proc. of the 2006 international workshop on service - oriented software engineering. New York, NY, USA:ACM,2006:74-80.
- [15] 李 熙,徐德智. 基于 WordNet 的概念语义相似度研究[J]. 湖南科技学院学报,2008,29(12):115-116.
- [16] 李涓子. 汉语词义排歧方法研究[D]. 北京:清华大学,1999.
- [17] 吴 芸,鱼 滨. 基于语义的 Web 服务匹配研究[J]. 计算机技术与发展,2007,17(3):16-19.
- [18] 于守健,何 丰,乐嘉锦. 基于接口匹配的 Web 服务自动组合[J]. 计算机科学,2007,34(3):61-68.
- [19] 李 涛,李涓子,王克宏. Web 服务异构消息匹配与重用[J]. 计算机学报,2006,29(7):1038-1046.

(上接第 62 页)