

文章编号:1001-9081(2010)09-2370-04

基于 SOA 软件架构的数据集成方法

郎 炯, 刘宴兵, 熊仕勇

(重庆邮电大学 计算机科学与技术学院, 重庆 400065)

(jiong000@163.com)

摘 要:根据数据的自治性和语义复杂性以及跨平台多样性等特点,并结合数字化业务平台的应用和需求,提出一种基于面向服务的体系结构(SOA)的数据集成方法。它是基于 JBI 规范,通过加入了四个服务模块解决了语义、多消息以及自治性等问题。最后结合 SOA 的相关技术和系统框架,并对其进行了实现。

关键词:面向服务的体系结构;数据集成;语义;Web 服务

中图分类号: TP311.5 **文献标志码:** A

Data integration method based on SOA software architecture

LANG Jiong, LIU Yan-bing, XIONG Shi-yong

(College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: According to the autonomy, semantic complexity and diversity of cross-platform of data, as well as combining the applications and requirements of digital business platform, a data integration method based on Service Oriented Architecture (SOA) software architecture was proposed in this paper. The proposed method, which was on the basis of Java Business Integration (JBI), could solve the problems of semantic aspect, multiple messages, and autonomy issues by means of adding four service modules. Finally, the proposed method was implemented by combining the relative techniques and system architecture of SOA.

Key words: Service Oriented Architecture (SOA); data integration; semantic; Web service

0 引言

随着数据库技术的快速发展,数据库系统的异构特性,造成数据的大量冗余和语义问题。跨平台数据库的运用对数据的自治性要求更高^[1]。针对这一问题,很多研究提出了自己的观点和解决方案。目前,关于数据集成体现结构主要有三种,即模式集成、数据复制和面向服务的体系结构(Service Oriented Architecture, SOA)^[2-3]。前两种是传统的集成方法,它在一定程度上解决了数据集成的语义和模型等问题,但灵活性和数据的自治性比较差^[4-5]。而面向服务体系结构的数据集成,使系统的灵活性和自治性增强。因此,成为当前研究的热点。

针对目前面向服务架构的异构数据集成技术中,客户端平台发送消息形式的多样化以及分布式数据源语义集成度不够高。本文提出一种基于 SOA 软件架构的数据集成方法。该架构是以业务驱动服务为原则,利用企业服务总线(Enterprise Service Bus, ESB)对集成服务进行动态管理,并利用应用服务层所提供的四个模块,可以实现多种平台消息的查询访问和众多数据模型的有效集成,并较好地解决了语义异构等问题,同时为用户提供的访问服务提供了有力的支持。

1 相关技术

1.1 SOA 概念和特点

SOA 是一种以服务为核心,业务被划分为一系列粒度的业务服务和流程服务。业务服务之间相对独立、可重用,由一个或多个分布式系统所实现。它主要有三个基本特点:松散耦合、粗粒度、位置和传输协议透明^[6]。服务之间的松散耦

合要求不同服务的功能不要相互依赖,一个服务应该能够自己实现所提供的接口功能,并且不依赖其他服务。它通过 SR (Service Registry) 和 ESB 来支持动态查询和管理。它要求服务之间的交互是动态的,这使得服务的提供者和请求者之间是高度解耦的,因此具有高度灵活性,并且一个服务内部结构变化不会影响其他服务。

1.2 Web 服务技术

Web 服务是以 XML 为基础,开放性的 Web 技术,是实现 SOA 的主要手段^[7]。它采用松散耦合的组织方式,使系统能够快速灵活地绑定到应用程序当中,是目前企业比较理想的集成方案。如图 1 所示,它是 Web 服务的一种典型的体系结构。它包含服务提供者(Service Provider, SP)、服务注册中心(Service Registry, SR)三个角色。服务提供者可以发布自己的服务,并且对使用自身服务的请求进行响应。服务注册中心用于对服务进行分类,并发布服务和提供服务搜索功能。服务请求者可在服务注册中心查找所需的服务,并使用服务。这种方式使得服务之间松散耦合,灵活性很强。



图1 Web 服务体系结构

2 数据集成体系结构

由于在 Internet 中存在大量的异构系统,而分布式的数据

收稿日期:2010-03-16;修回日期:2010-05-05。

基金项目:重庆市科技攻关计划项目(2009AB2053);重庆市教育委员会自然科学基金资助项目(KJ080505)。

作者简介:郎炯(1983-),男,四川达县人,硕士研究生,主要研究方向:数据集成;刘宴兵(1971-),男,四川遂宁人,教授,博士,主要研究方向:无线网络、网格计算;熊仕勇(1976-),男,四川自贡人,工程师,硕士,主要研究方向:软件架构。

源要求数据有很强的自治性。它们要求不用通知集成系统的前提下,就可以改变自身的结构和数据,同时也要满足数据的实时性要求。而基于 SOA 的数据集成把各种数据源,以各种不同粒度的服务进行了封装。服务动态性的交互使得服务之间可以任意访问和组合,因此 SOA 有强大的灵活性和动态性,这也使得对企业数据的共享和集成带来了便利。

2.1 JBI 规范

JBI(Java Business Integration)^[8]是 SUN 公司解决 SOA 的方案,它通过建立一种可以集成各种组件的运行环境并为 JBI 组件和组件周期的管理提供了一套服务。它使用标准的服务描述语言来描述插件组件间基于消息的服务调用达到组件之间的交互,这种方式为组件所提供和消费的服务提供了统一的模型。

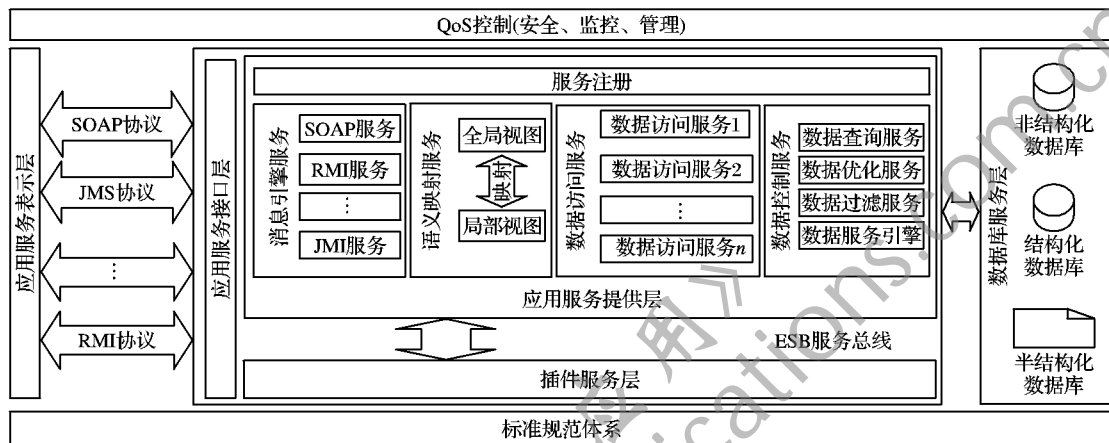


图2 基于 SOA 的数据集成框架

2.3 数据集成服务模块

应用服务提供层是本框架的核心,本文所提出的体系结构,通过在应用服务层加入四个模块服务,独立处理数据集成。如图2所示,它由消息引擎服务、语义映射服务、数据访问服务和数据控制服务等四个模块组成。在图2中,请求由应用服务表示层发出,该表示层由多种客户端异构平台组成,且每个平台可发送自己独立的消息,消息对数据信息进行了封装。消息一旦发出就交给消息映射引擎模块处理。该模块由抽象消息对象、具体的消息对象、抽象消息工厂对象、具体的消息工厂对象等4个对象组成,它根据消息的格式,判断消息类型,并由消息对象对消息进行分解和重组,把消息转换成标准消息交由语义映射服务模块处理。在语义映射服务中,这种标准消息是基于全局视图的。该模块利用网络本体语言(Web Ontology Language, OWL)语言对全局视图和局视图部行描述,并对视图中的各种待查询的信息进行定义,建立匹配规则。根据匹配规则,生成一种本体映射文件,此文件由一系列相似信息的集合组成,查询的请求可根据映射文件,把全局查询分解成可执行的子查询,此语义处理详细过程将在系统的设计和实现部分阐述。数据访问服务模块根据子查询和对应的参数信息,初始化 RowSet 对象(一种存储数据的对象),并加载数据模型,执行查询。根据数据模型的多少,RowSet 对象可以有多个,且存储了查询的结果。数据控制服务再从 RowSet 对象中获取数据,并进行冗余处理,最后生成统一的 XML 格式的集成数据。

该架构包含三个要素:安全、监护、管理。以上各层的建设,需要依托现有的法律、法规及业务标准和规范才可以成功运行。该方法解决了多个平台消息之间的异构问题,也解决了分布式数据库存在的多种语义问题。它满足了以业务驱动服务,以服务驱动技术的 SOA 设计原则。利用 ESB 服务总线的管理,系统的安全和性能也得到了大大提高。

2.2 数据集成框架

本文所提出的数据集成框架,按照 JBI 规范,由数据库服务层、应用服务提供层、插件服务层、应用服务接口层和应用服务表示层五层组成。如图2所示,用户由应用服务表示层发送查询请求消息,服务器接收消息,并做出处理。应用服务接口层以接口的形式提供服务给应用服务表示层,表示层获取服务信息,并根据获取的信息从应用服务提供层提供的服务实现动态的绑定。绑定后的服务可根据消息的请求进行消息转换、语义映射和查询处理,并从数据库服务层获取集成数据。插件服务层作为体系结构的可扩展式的层次结构。该层把所有的可扩展功能都集中在这个层次,作为结构的补充和增强。

3 系统的设计与实现

下面结合数字业务支撑系统的开发,来说明本文所设计的体系结构的实现方式。随着电子商务应用范围的扩大和深入,商务系统具有更复杂的结构和更高的要求,因此,电子交易系统作为一个运行在广域网中的分布式计算机系统,需要连接不同企业的服务器,不可避免会遇到大量的异构平台和语义问题。本文结合 SOA 框架,将使系统具有更好的跨平台性和灵活性。

3.1 系统的设计方法

基于 SOA 数字业务支持系统是基于服务软件开发设计,其中以低运作成本、短期开发时间、提高代码的可重复性为基本理念。如图3所示,它是数字业务系统支撑系统体系结构,各种模块组件采用松散耦合的方式进行组合。框架管理由业务组件共享资源,提供根本机制允许组件之间的交互。系统主要采用企业级 Java 组件(Enterprised Java Bean, EJB)技术,它是采用 Web 服务技术标准,并采用多种处理模式和以流程为中心的业务处理思想。

3.2 系统的流程结构

本系统主要由四种文档组成:JSP 文档、XML 文档、WSDL 文档和 javabean 文档。本系统采用 MVC 设计模式,由 servlet 来处理 *.do 请求,经处理后的请求可转发到相应的 JSP 页面。如图4所示。

系统的流程如下。

1) Query.jsp 提供了一种统一查询接口,它向服务器端发送 Query.do 查询请求。服务端接受到 Query.do 请求后,由 QueryServlet 处理集成查询请求,然后 QueryServlet 再发送 Server.do 请求,由 GetServerInfoServlet 处理,获取服务信息。

2) GetServerInfoServlet 从应用服务接口层(Server.wsdl),获得所有服务信息,然后在把服务信息返回给 QueryServlet。

3) QueryServlet 根据服务信息,从应用服务提供层所提供的服务,绑定所需要的查询服务。它先由客户端发送消息,然后由 MessageServer 处理,处理后的消息由 LanguageServer 进行语义映射生成可执行子查询,子查询由 SQLServer 处理,最后由 ControlServer 生成集成数据。

4) 集成数据用 Integration.XML 表示,最后应用服务表示层将获取的集成数据,将它显示到 Show.jsp 页面。

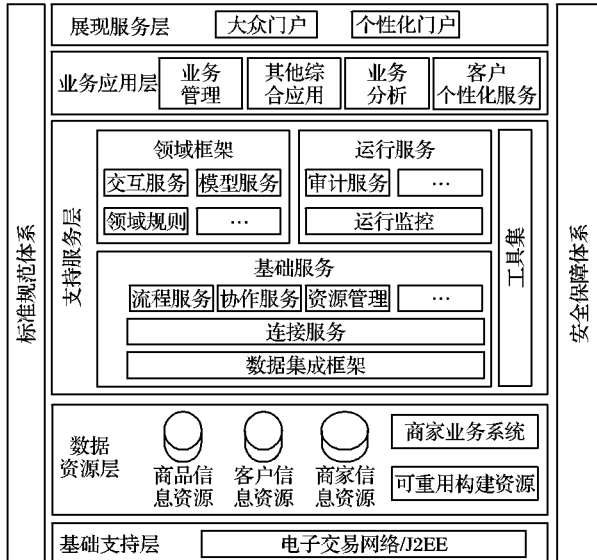


图3 数字化业务支持系统的体系结构

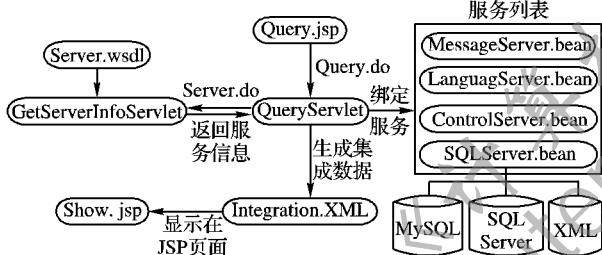


图4 数据集成系统的流程

3.3 系统的运行机制

3.3.1 消息创建

ESB 是整个系统框架的核心环境,它主要由若干服务组成。详细过程如下:首先创建消息对象,如图5所示,消息由客户端发送出来,由消息监听对象进行监听,并创建消息工厂。设置 Info_no 为消息类型序列号,消息工厂根据 Info_no 的序列号,创建不同的消息对象。消息对象的创建采用工厂设计模式。此设计模式含有4个角色:抽象消息对象、具体的消息对象、抽象消息工厂对象和具体的消息工厂对象。抽象消息对象含有一个 doMessage 方法,此方法用于处理接受的消息。而抽象消息工厂含有 Create Message,此方法用于创建消息对象。具体的消息对象和具体的消息工厂对象是抽象消息对象和抽象消息工厂的实例,它们继承抽象层的方法和属性。并由消息工厂创建消息对象,根据 Info_no 的值,创建对应的消息对象,且用 doMessage 处理消息,以统一一个格式封装数据,形成标准化消息。抽象对象的设计,有利于对消息机制的扩充,使系统更具有灵活性。

3.3.2 语义处理

标准化后的消息是基于全局视图的查询请求,根据映射规则,它把全局视图转化成局部视图,并分解成可执行的子SQL语句。每个子SQL语句对应不同的数据模型。此过程是基于惠普实验室的开放源代码 jena Semantic Web Framework^[9]的框架作为基础,具体的映射匹配过程如图6所示,其语义映射匹配方法包含4个主要过程:1)读取 OWL 映

射文件遍历类节点;2)获得类节点的所有属性,匹配关键字信息;3)遍历类节点的子类节点,匹配关键字信息;4)获得关键字的等价类获得等价属性,分解全局概念,得到的分查询进行存储。

最后将执行分查询,合并结果集。

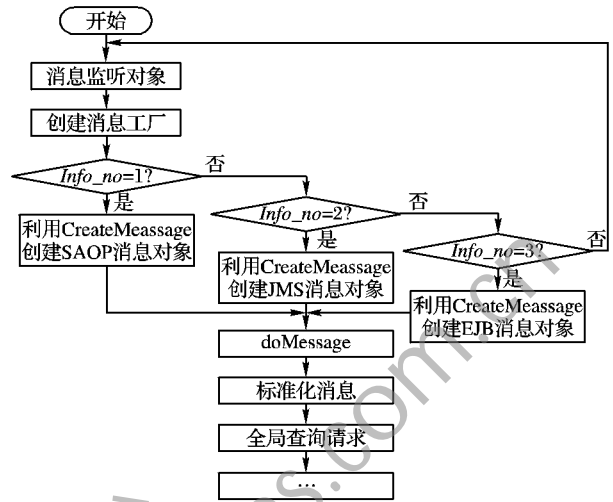


图5 消息创建流程

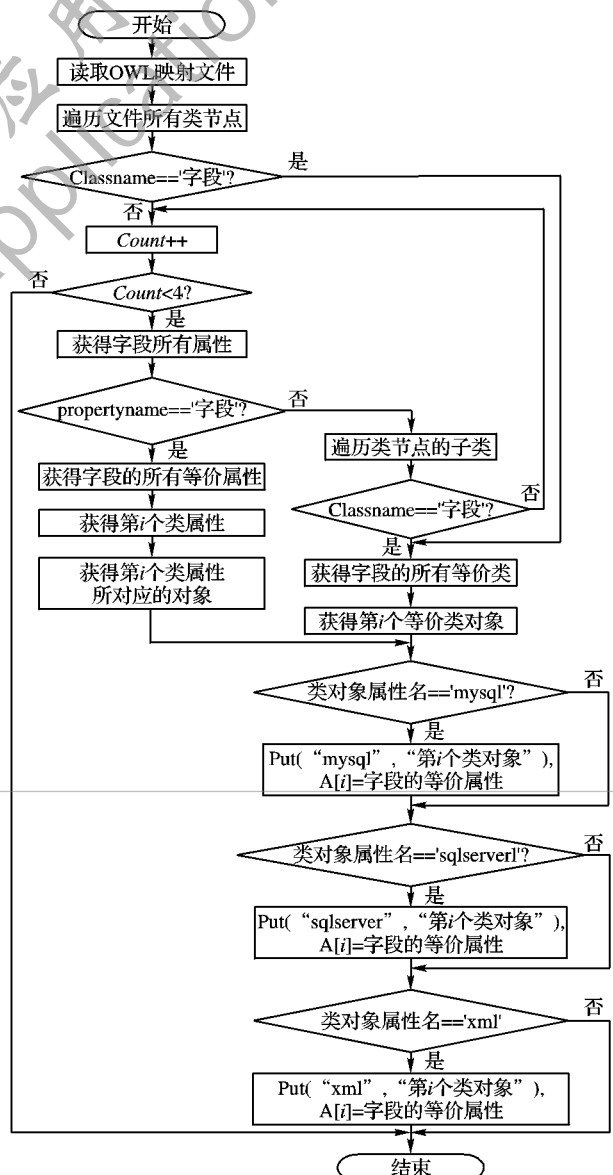


图6 语义处理流程

4 系统的测试

本数据集成系统是数字业务支撑系统的一个小的模块,位于图3所示的资源层,该系统基于上述框架,实现了三种分布式数据源的查询。该实例由重庆某大型网络游戏公司的数字业务支撑系统中的10000条网游客户信息为例,此信息由MySQL、SQLServer、XML三种数据源所存储。它包含了结构化和半结构化两种数据模式,具有代表性。为了便于测试,本系统只简单提供了网游客户信息的查询结果界面,并只给出了部分网游客户信息。系统的查询结果,如图7所示。该系统主要查询网游客户的订单信息,该信息由5个字段组成,包含订单号、充值账户、订单IP、交易金额和订单时间。为了区别各个数据源所存储的位置的差异性。在页面中,用“数据源”这个字段来表示各种商品信息的存储位置。从查询结果可以看出,连续的订单号属于一种类型的数据模型,因此基于SOA软件架构的数据集成系统能很好保证数据的自治性和数据的共享,它屏蔽了底层数据的访问细节,使用户能透明访问各种分布式数据源。

本系统是基于SOA的软件架构框架开发而来,它是在jena基础上构造的,运行在Tomcat服务器下,并把消息引擎服务、语义映射服务、数据访问和数据缓存服务部署到ESB服务总线中,使用RowSet对象来加各种模型的数据库,用XML来交换数据格式,实现了混合数据查询。它通过抽象层次框架的设计,构造出了更高层次的抽象服务,该系统完全采用了SOA思想,设计出新的框架结构,较好地集成了分布式数据源,因此系统有好的灵活性和可扩展性。

—XML、SqlServer、Mysql(三种数据源的集成实例)—

订单号	充值账户	订单IP	交易金额	订单时间	数据源
164	sk1239796	216.010.164.200	46	2009/01/05 02:37:05	SqlServer
185	slarjle018	201.146.250.094	47	2009/01/05 04:22:46	SqlServer
196	q4157637688	222.140.003.126	28	2008/01/05 05:02:58	SqlServer
197	weid520qun	258.209.000.251	66	2009/01/05 05:09:22	SqlServer
199	yt23458502	201.185.062.162	35	2009/01/05 09:55:11	SqlServer
4324	9enllng2005658	201.185.082.162	41	2009/01/05 10:42:39	XML
4325	yy2871461200081	185.082.162	52	2009/01/023 05:53:26	XML
4326	fxnongit	125.063.031.171	55	2009/01/023 06:08:08	XML
4327	hu66145912	222.212.073.113	43	2009/01/023 06:31:31	XML
6322	zsf1874	858.017.014.144	28	2009/01/023 07:06:04	XML
6323	z4561455	125.086.181.138	87	2009/01/023 20:41:40	Mysql
6324	mxda2007	222.093.173.069	40	2009/01/023 20:47:26	Mysql
8325	czhenly	125.085.110.209	34	2009/01/023 20:59:56	Mysql

图7 系统查询结果

5 结语

本文提出一种基于SOA软件架构的数据集成方法,通过消息服务引擎处理多消息异构平台消息的查询访问,避免了传统平台多样性而不能处理其他消息的场景。语义映射服务模块的设计,可独立解决数据语义异构问题,保证了数据源的自治性。另外数据访问服务和数据控制服务,使系统支持多种数据模型和海量数据的查询,并可使集成的数据作二次优化。最后利用ESB对服务进行动态管理,屏蔽了用户访问服务的细节。从本文所给出方案中可以看出,体系结构能较好地解决数据集成的跨平台和部分语义问题,但对数据安全控制和异构数据源的语义还需要进一步研究。如何为系统的数据集成提供更好的语义和数据安全的控制是今后需要进一步的研究。

参考文献:

- [1] WANG J, YU A R, ZHANG X Y, *et al.* A dynamic data integration model based on SOA [C]// ISECS International Colloquium on Computing, Communication, Control, and Management. Sanya: [s. n.], 2009: 196–199.
- [2] NI Y H, WANG H B, HUANG N S, *et al.* A heterogeneous system integration framework for business collaboration [C]// IEEE International Conference on Intelligent Computing and Intelligent Systems. Shanghai: [s. n.], 2009: 217–221.
- [3] JIN Y, SHIDIACATTA S. A framework of fuzzy triggers for XML database systems [C]// IEEE International Conference on Information Ruse and Integration. Las Vegas: [s. n.], 2009: 296–299.
- [4] 刘伟, 孟晓峰, 孟卫一. Deep Web 数据集成研究综述[J]. 计算机学报, 2007, 30(9): 1476–1486.
- [5] 陈跃国, 王京春. 数据集成综述[J]. 计算机科学, 2004, 31(5): 48–51.
- [6] ERL T. SOA principles of service design [M]. London: Prentice Hall, 2007.
- [7] SI N, YU P F, ZHANG L. Semantic SOA based integrated development environment for complex information systems [C]// Proceedings of the 7th International Conference on System Simulation and Scientific Computing. Beijing: [s. n.], 2008: 1521–1526.
- [8] BADURA A, SAKOWICZ B, MAKOWSKI D. Integration of management protocols based on Apache ServiceMix JBI platform [C]// 10th International Conference on Experience of Designing and Application of CAD Systems in Microelectronics. Polyana-Svalyava, U-KRAINE: [s. n.], 2009: 381–384.
- [9] McCARTHY P. Jena [EB/OL]. [2009–08–12]. <http://www.ibm.com/developerworks/cn/java/j-jena/>.

(上接第2347页)

- [6] HASHEMIAN S V, MAVADDAT F. A graph-based framework for composition of stateless Web services [C]// ECOWS'06: Proceedings of the European Conference on Web Services. Washington, DC: IEEE Computer Society, 2006: 75–86.
- [7] THAKKER D, OSMAN T, AI-DABASS D. Knowledge-intensive semantic Web services composition [C]// UKSIM'08: Proceedings of the Tenth International Conference on Computer Modeling and Simulation. Washington, DC: IEEE Computer Society, 2008: 673–678.
- [8] YE L, ZHANG B. Discovering Web services based on functional semantics [C] // APSCC'06: Proceedings of the 2006 IEEE Asia-Pacific Conference on Services Computing. Washington, DC: IEEE Computer Society, 2006: 348–355.
- [9] AKKIRAJU, SRIVASTAVA R, IVAN B, *et al.* Semantic matching to achieve Web service discovery and composition [C]// Proceedings of the 8th IEEE International Conference on and Enterprise Computing, E-Commerce, and E-Services. Washington, DC: IEEE Computer Society, 2006: 70–78.
- [10] GAMHA Y, BENNACER N, BEN ROMDHANE L, *et al.* A state-chart based model for the semantic composition of Web services [C]// SCW'07: 2007 IEEE International Conference on Services Computing-Workshops. Washington, DC: IEEE Computer Society, 2007: 49–56.
- [11] LI H, WANG H Y, CUI L. Automatic composition of Web services based on rules and meta-services [C]// CSCWD'07: Proceedings of the 11th International Conference on Computer Supported Cooperative Work in Design. Berlin: Springer, 2007: 496–501.
- [12] SIRIN E, PARSIA B, WU D, *et al.* HTN planning for Web service composition using SHOP2 [J]. Journal of Web Semantics, 2004, 1(4): 377–396.
- [13] THIAGARAJAN R, STUMPTNER M. Service composition with consistency-based matchmaking: A CSP-based approach [C]// ECWOS'07: Proceedings of the Fifth European Conference on Web Services. Washington, DC: IEEE Computer Society, 2007: 23–32.