

基于 OWL-S 和 HTN 的 Web 服务组合 原型系统的研究

吴玲娟, 胡文江, 高永兵, 赵 菲

(内蒙古科技大学 信息工程学院, 内蒙古 包头 014010)

摘 要: Web 服务是近年发展的一种分布式技术, 引起了越来越多研究者的关注。随着网络上 Web 服务技术描述的信息系统数量的不断增加, 将现有的业务流程进行组合以满足较为复杂的业务系统成为当前发展的主要趋势。此外, 由于 Web 服务所处环境的复杂性和动态性, 使得 Web 服务组合任务变得非常困难。因此, Web 服务组合逐渐成为语义 Web 服务领域研究的热点和难点。描述了一个基于 OWL-S 和 HTN 的 Web 服务组合框架, 并详细介绍了如何利用 HTN 规划解决 Web 服务组合问题的算法思想。通过学生选课服务实验, 验证了该方法的可行性。

关键词: Web 服务; OWL-S; HTN; Web 服务组合; HTN 规划

中图分类号: TP392

文献标识码: A

文章编号: 1673-629X(2011)01-0053-05

Research of Composition Prototype System of Web Services Based on OWL-S and HTN

WU Ling-juan, HU Wen-jiang, GAO Yong-bing, ZHAO Fei

(School of Information Engineering, Inner Mongolia University of Science & Technology, Baotou 014010, China)

Abstract: Web services is a distributed technology and have developed in recent years, paid more and more attention to researchers. As the increasing number of information systems in the network described by Web services technology, the combination of existing business processes to meet the more complex ones has become the main trend of development. In addition, because of the complexity and dynamic nature of the Web services environment, Web service composition makes it a very difficult task. So Web service composition has become a hot and difficulty in the field of semantic Web services. This paper describes a Web service composition framework based on OWL-S and HTN and details the algorithm idea of how to use HTN planning to solve the problem of Web service composition. Finally, the course selection service of students is implemented to demonstrate the feasibility of the method.

Key words: Web services; OWL-S; HTN; Web services composition; HTN planning

0 引言

Web 服务的目标就是利用一系列的技术来充分发挥它的潜力^[1]。随着 Internet 的发展及现实的需要, 网络(Web)服务的数量也在飞速地增加。因而, 如何利用 Web 服务组合技术准确地向用户提供合乎功能需求的组合服务是当今研究的热点。

一般来说, 服务组合分为两种方法: 静态和动态方法^[2]。在静态组合的情况下, 参与组合的服务在设计阶段就被选择^[3]。而在动态组合的情况下, 参与组合

的服务在运行期间才被选择^[4]。随着信息系统日益复杂和网络上的不确定和多变性, Web 服务的动态组合逐渐成为当前的发展趋势。

文献[5]提出一种面向用户的 Web 服务组装机方法, 将组装过程分为需求拆分、服务选择、执行服务组装和服务结果合并四个阶段, 并提出了基于组装模式的用户需求输入及拆分方法、基于 QoS 的服务选择模型和基于中间模式的异构消息转换模型。文献[6]给出了一种基于领域本体和 OWL-S 的 Web 服务动态组合方法, 它将 Web 服务从 WSDL 映射到 OWL-S, 并在领域本体的帮助下对 OWL-S 服务进行语义标注, 从而统一 Web 服务中的领域概念, 并通过定制 OWL-S 组合过程, 实现对 Web 服务的组合。文献[7]提出一个基于语义的 Web 服务动态组合框架, 它从体系结构上主要划分为四层, 即数据存储层、组合服务层、语义推理层和应用表示层。其中, 组合服务层中

收稿日期: 2010-04-28; 修回日期: 2010-07-03

基金项目: 内蒙古自然科学基金项目(200711020813); 内蒙古自治区高等学校科学研究项目(NJC07109)

作者简介: 吴玲娟(1985-), 女, 河北邢台人, 硕士研究生, 研究方向为语义 web 服务; 胡文江, 硕士生导师, 教授, 研究方向为智能搜索技术; 高永兵, CCF 会员, 硕士生导师, 副教授, 研究方向为数据库与 XML。

组件的作用就是将一个用标准 XML 查询语言翻译成能在经过封装的数据源上执行的操作,抽取和打包查询结果到一个 XML 文档,最后将该文档返回给上层调用。我们采用分层任务网络 HTN(Hierarchical Task Network)来实现 Web 服务动态组合,即将服务组合看作一个规划问题。理想情况下,给定用户目标和 Web 服务集合,规划器需要发现满足该目标的 Web 服务组合。我们相信 HTN 规划非常有希望满足该目的,因为在 HTN 规划中任务分解概念与 OWL-S 过程本体中复合过程分解概念很相似。本文说明如何使用基于 HTN 的 JSHOP2 规划系统来实现 OWL-S 描述的 Web 服务的自动组合。

1 相关知识

1.1 OWL-S

根据文献[8], OWL-S 包含了三个上层本体,分别是 SERVICEPROFILE、SERVICEMODEL、SERVICEGROUNDING。其中,在 Web 服务组合过程中,OWL-S 的过程本体 SERVICEMODEL 最为重要。它包括:原子过程(AtomicProcess),复合过程(CompositeProcess)和简单过程(SimpleProcess)。原子过程没有子过程,一步执行,它可以直接被调用来完成某个任务。执行一个原子过程相当于调用相应的 Web 可访问程序,将它的输入参数绑定特定的值。一个组合过程代表一个复合 Web 服务,也就是说,它能分解成其他的原子过程、简单或复合过程。而一个组合过程的分解通过它的控制结构来指定。控制结构主要包括 Sequence, Unordered, Choice, If-Then-Else, Iterate, Repeat-Until, Repeat-While, Split and Split+Join。简单过程是不可调用的,也没有对应的 Grounding,但是它们和原子过程一样,被认为只有一个执行步骤。简单过程一般作为元素的抽象,它们或者提供了一些原子过程的视图,或者是对组合过程的简化的表达。在前一种情况下,简单过程被认为是 realizedBy 原子过程;在后一种情况下,简单过程被认为是 expandsTo 复合过程。

1.2 JSHOP2

HTN 规划是一种采用分层分解的思想,即将一个复杂的任务递归的分解成一系列较小的子任务,直到可以用规划操作直接执行的原子任务为止。SHOP2 是一个独立于领域的 HTN 规划系统,在 2002 年的 14 种规划器国际规划比赛中荣获前四名之一^[9]。它与大多数其他 HTN 规划系统的一个不同点是在 SHOP2 的同样顺序的任务规划将被较晚地执行,其目的是使得这些任务在规划过程中能够了解世界的当前状态,从而可以让 SHOP2 的前提条件评估机制具有强大的推理能力和调用外部程序的能力。SHOP2 的这种思想

为与外部信息资源(包括 Web 资源)的集成规划奠定了基础。JSHOP2 是 SHOP2 的 Java 实现版本。

为了在给定的领域进行规划,规划器需要知道该领域的知识信息。JSHOP2 的知识库包括算子和方法(外加各种非行为的相关事实和公理)。其中,每个算子描述了为了完成某个原子任务需要干什么,而每个方法告诉怎样将某个复合任务分解成一系列有序的子任务。

定义 1(算子):一个算子表示为一个四元组 $(h(v), Pre, Del, Add)$, 这里, $h(v)$ 是一个输入参数为 v 的原子任务; Pre 表示操作的前提条件; Del 表示删除那些执行操作后状态列表中不再正确的状态。 Add 表示把执行操作后的新状态添加到状态列表中。

定义 2(方法):方法可表示为形式如 $(h(v), Pre1, T1, Pre2, T2, \dots)$, 其中 $h(v)$ 是一个拥有参数列表为 v 的复杂任务,每个 Pre_i 是一个前提条件表达式, T_i 表示有序的部分子任务集合。

定义 2 与一个条件表达式相似:它告诉 JSHOP2, 如果满足 $Pre1$, 那么使用 $T1$; 否则, 如果 $Pre2$ 被满足, 那么使用 $T2$ 等等。一个任务列表只是说明了这个复合任务如何分解成子任务的。在列表中的任务可能是原子任务,也可能是复合任务,而且一个任务列表可能是有序的,也可能是无序的。

定义 3(规划问题):一个规划问题可定义为一个三元组 (S, T, D) , 其中 S 表示初始状态, T 表示任务列表, D 表示一个领域描述。 JSHOP2 将 (S, T, D) 作为输入, 返回一个规划 $P = (p1, p2, \dots, pn)$ 。

2 系统框架

本节主要介绍基于 OWL-S 和 HTN 的 Web 服务组合框架,如图 1 所示。该框架主要包括三个部分,分别是服务发现模块、组合模块和执行模块。其中,服务组合模块是该框架的核心部分,它负责将 OWL 描述的领域本体和 OWL-S 描述的服务本体分别转化为规划器可以执行的问题描述文件和领域描述文件。下节将主要介绍 JSHOP2 是实现 Web 服务的组合的过程。

3 接口匹配算法

注册服务器端将客户端(服务发现模块)传来的关于服务能力的描述服务请求者与本地存储的所有语义 Web 服务的能力描述进行语义匹配,并得到的匹配结果列表返回给客户端。在服务匹配过程中,主要涉及以下五个参数:领域本体基本 URI、服务分类、服务产品、输入和输出。如果用户输入的服务能力描述文件中 URI 与注册服务器端保存的 OWL-S 文件导入的领域本体 URI 相同,则匹配结果为 EXACT, 否则为

FAIL。具体算法如下:

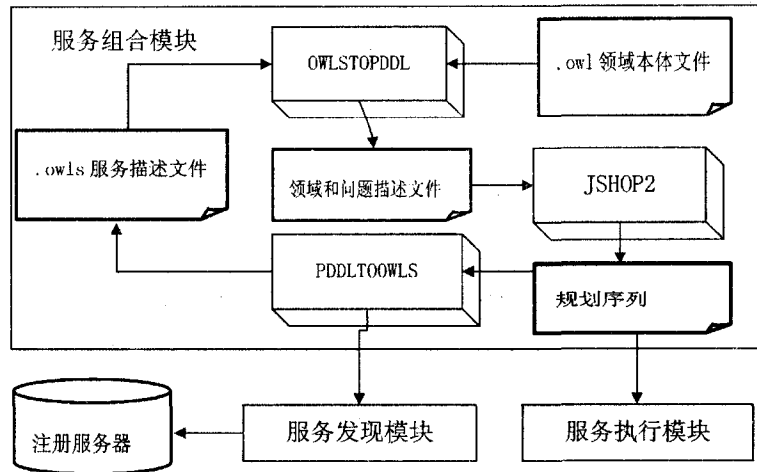


图 1 服务组合框架

Match(ReqConcept R, AdvtConcept A)

```

if A = R then return EXACT;
if R subclassOf A
then return SUBSUME;
if A subclassOf R
then return RELAXED;
otherwise fail;

```

4 从 OWL-S 到 JSHOP2 的转化

在本节中,详细说明如何将一个复合过程的组合问题编码成一个 JSHOP2 规划问题,从而使用 JSHOP2 和 OWL-S 描述 Web 服务来自动产生一个 Web 服务调用的组合。

给定一个 OWL-S 过程模型集合 $K = \{K_1, K_2, \dots, K_n\}$, 每个过程模型满足以下两个条件:第一,在 K 中定义的所有原子过程或者拥有 effects 或者 outputs,但是不能同时带有这两个。根据基于情景演算的 OWL-S 语义^[10], outputs 描述了执行 Web 服务后的知识效果,而 effects 描述了执行 Web 服务后的世界的物理状态。为了能够在规划中从 Web 服务中收集信息,要求原子过程或者只是提供信息或者只是改变世界状态。第二,在 K 中没有带 OWL-S 的 Split 和 Split+Join 控制结构的复合过程。目前 JSHOP2 不能处理并非操作。因此在转化过程中,只考虑不带并发控制结构的 OWL-S 过程模型。

定义 4(OWL-S 服务组合):已知一个 OWL-S 过程模型集合 $K = \{K_1, K_2, \dots, K_n\}$, 且 K_i 满足以上两个假设, S_0 为初态,给定一个需求 R , 返回一个服务序列 $P = (P_1, P_2, \dots, P_n)$, 其中, $P_i \in K$, 且 P 为一个复合过程。

将 OWL-S 过程定义集合 K 转化为 JSHOP2 的领域 D 过程包括三种情况:将 K 中每个带 effects 的原子过程,将其编码为 JSHOP2 的一个模拟改变世界状态的 Web 服务执行结果的算子;将 K 中每个带 outputs 的

原子过程,将其编码为 JSHOP2 的前提条件包括对提供信息 Web 服务的一个调用的算子;将 K 中每个简单或复合过程编码为一个或多个 JSHOP2 方法,这些方法将告诉如何将代表该简单或复合过程的 HTN 任务进行分解。

下面是将一个 OWL-S 定义的只带 effects 的原子过程转化为一个 JSHOP2 算子的伪代码,如下:

```

TRANSLATE-ATOMIC-PROCESS-EFFECT(Q)
Input: a OWL-S definition Q of an atomic process A
with only effects
Output: aJSHOP2 operator O
Procedure:
(1) v = the list of input parameters defined for A as

```

- in Q;
- (2) Pre = a conjunct of all the preconditions of A, as defined in Q;
- (3) Add = collection of all positive effects of A, as defined in Q;
- (4) Del = collection of all negative effects of A, as defined in Q;
- (5) Return O = (A(v) Pre Del Add);

以上算法转化到的算子通过改变它的本地状态来模拟一个改变客观世界状态的 Web 服务,且这样的 Web 服务在规划期间将从来不被执行。

同理,将一个 OWL-S 定义的只带 outputs 的原子过程转化为一个 JSHOP2 算子的算法,这样得到的算子将根据它的前提条件调用只提供信息的 Web 服务,因此,这样的 Web 服务在规划期间被执行。关于这些原子服务的算子完全是“薄记的”,将不会出现在最终的规划中。

下面说明如何将一个 OWL-S 定义的复合过程转化为一个 JSHOP2 的方法的过程。现以带 Sequence 控制结构的复合过程为例,算法如下:

```

TRANSLATE-Sequence-PROCESS(Q)
Input: a OWL-S definition Q of a composite process
C with Sequence control construct.
Output: a JSHOP2 method M.
Procedure:
(1) v = the list of input parameters defined for C
in Q.
(2) Pre = conjunct of all preconditions of C as defined
in Q.
(3) B = Sequence control construct of C as defined in
Q.
(4) (b1, b2, ..., bm) = the sequence of component
processes of B as defined in Q.
(5) T = ordered task list of (b1, b2, ..., bm).
(6) Return M = (C(v) Pre T).

```

以此类推,可以得出将一个 OWL-S 过程模型集

合转化为一个 JSHOP2 领域 D 的算法:

TRANSLATE-PROCESS-MODEL(K)

Input: a collection of OWL-S process models K.

Output: aJSHOP2 domain D.

Procedure:

(1) D = ;

(2) For each atomic process definition Q in K

If this atomic process has only outputs

O = TRANSLATE-ATOMIC-PROCESS-OUTPUT

(Q)

If this atomic process has only effects

O = TRANSLATE-ATOMIC-PROCESS-EFFECT

(Q)

add O to D

(3) For each simple process definition Q in K

M = TRANSLATE-SIMPLE-PROCESS(Q)

add M to D

(4) For each composite process definition Q in K

If the process has a Sequence control construct

M = TRANSLATE-Sequence-PROCESS(Q)

If the process has a If-Then-Else control construct

M = TRANSLATE-If-Then-Else-PROCESS(Q)

If the process has a Choice control construct

M = TRANSLATE-Choice-PROCESS(Q)

If the process has a Repeat-While control construct

M = TRANSLATE-Repeat-While-PROCESS(Q)

If the process has a Repeat-Until control construct

M = TRANSLATE-Repeat-Until-PROCESS(Q)

If the process has a Unordered control construct

M = TRANSLATE-Unordered-PROCESS(Q)

add M to D

(5) Return D

为了简单起见,以上伪代码没有考虑组合过程的子过程嵌套问题。在实际应用情况中,经常是把复合过程中每个控制结构都看做一个复合过程,然后对每个复合过程进行递归。

根据文献[11]可得到 JSHOP2 进行规划的简化算法。该算法利用了 HTN 的一个优点,即在不考虑如何分解它的子任务的情况下方法可以被写。

Procedure JSHOP2(s, T, D)

If T is empty then return empty plan

Let t be the first task in T

If t is a primitive task then

Find an operator o = (h Pre AddDel) in D such that h unifies with t and s satisfies Pre

If no such o exists then return failure

Let s' be s after deleting Del and adding Add

Let T' be T after removing t

Return[o, JSHOP2(s', T', D)]

Else if t is a composite task

Find a method m = (h Pre1 T1 Pre2 T2 ...) in D such that h unifies with t

Find the task list Ti such that s satisfies Prei and does not satisfy Prek, k < i

If no such Ti exists then return failure

Let T' be T after removing t and adding all the elements in Ti at the beginning

Return JSHOP2(s', T', D)

End if

End JSHOP2

5 注册服务器的实现

为了实现客户端与服务器之间的通信,这里采用的 Hessian 远程处理方案。Hessian 是基于 HTTP 的瘦的二进制协议,即轻量级远程管理协议^[12]。语言无关但主要用于 Java-to-Java 的远程处理。在进行基于 Hessian 的课题开发时,必须满足以下两个要求:

(1) 在注册服务器端的配置步骤:包含的 Hessian.jar 包;设计一个接口,用来给客户端调用;实现该接口的功能;配置 web.xml 文件和相应的 servlet 对象必须实现 Serializable 接口。

(2) 客户端也要包含 Hessian.jar 的包,利用 HessianProxyFactory 调用远程接口。

6 Web 服务组装实例

现以学生选课组合服务为例。首先,使用 protégé 本体工具构建关于学生选课的领域本体;其次,组合框架通过导入所建领域本体,然后对其加载;然后,组合框架便根据 TRANSLATE-PROCESS-MODEL(K) 算法对应的一个 OWL-S 过程模型集合转化为相应的 JSHOP2 领域 D,之后, JSHOP2 便调用 Procedure JSHOP2(s, T, D) 规划算法进行规划,从而获得最后的规划序列直接执行,学生选课组合服务完成。图 2 为运行结果。

7 结束语

文中主要描述了一个基于 OWL-S 和 HTN 的 Web 服务组合框架,并详细介绍了如何利用 HTN 规划解决 Web 服务组合问题的算法思想。该框架主要由三个部分组成,分别是服务发现模块、组合模块和执行模块。服务组合模块主要负责将 OWL 描述的领域本体和

OWL-S 描述的服务本体分别转化为规划器可以执行的问题描述文件和领域描述文件,以使 JSHOP2 可处理。而且在使用规划算法的基础上同时还提供将组合的服务经过 OWL-S 语义标注后进行发布,以增强本系统的组合能力。

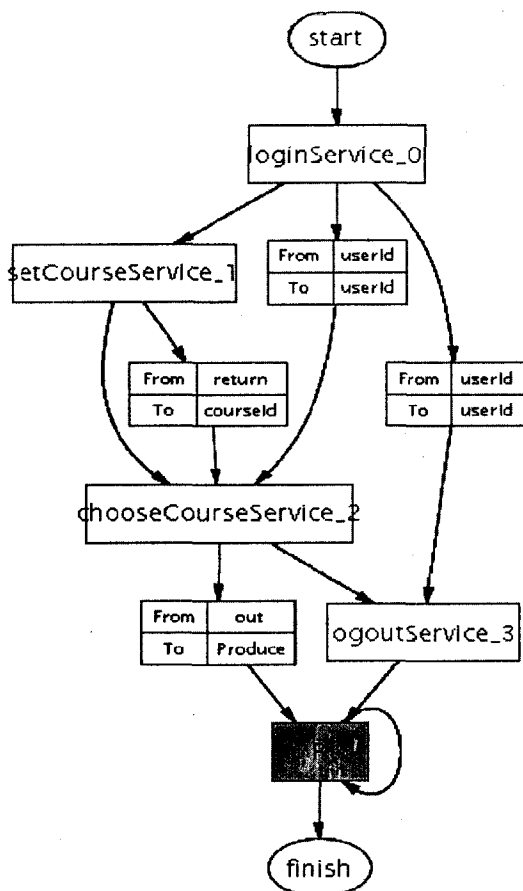


图 2 语义 Web 服务组合器运行结果

本系统在服务匹配过程中未考虑 Web 服务的非功能 Qos 因素,比如服务费用、服务的响应时间、服务的可用性等。它可以与领域本体匹配算法相结合,在首次匹配结果为 Exact 或 Subsume 的情况下,通过 Qos 约束,可获得更准确的、符合用户需求的 Web 服务。这是本课题的下一步研究工作。

参考文献:

- [1] W3C. Semantic Web Activity [EB/OL]. 2006. <http://www.w3.org/2001/sw/>.
- [2] Liu Xingwei, Zhao Hui. An AI Planning Based Approach for Automated Web Services Composition [C]//The 2007 International Conference Proceedings on Intelligent Systems and Knowledge Engineering (ISKE2007). Chengdu: [s. n.], 2007.
- [3] Koehler J, Srivastava B. Web service composition: current solutions and open problems [C]//The 2003 International Conference on Automated Planning and Scheduling (ICAPS'03). Trento, Italy: [s. n.], 2003.
- [4] Skogan D, Gronmo R, Solheim I. Web service composition in UMLm [C]//The 8th International IEEE Enterprise Distributed Object Computing Conference (EDOC). Monterey, California: [s. n.], 2004.
- [5] 李鹏, 战德臣, 刘国忠, 等. 一种面向用户的 Web 服务组装方法 [J]. 计算机应用, 2009, 29(11): 3120-3123.
- [6] 吴善明, 沈建京, 韩强. 基于领域本体和 OWL-S 的 Web 服务组合方法 [J]. 计算机工程, 2009, 35(21): 256-257.
- [7] 郑娅峰, 鱼滨. 基于语义 Web 的动态组合服务关键技术研究 [J]. 计算机工程与应用, 2005(3): 45-48.
- [8] Martin D, Burstein M, Hobbs J, et al. OWL-S: Semantic Markup for Web Services [EB/OL]. 2004. <http://www.w3.org/Submission/OWL-S>.
- [9] Sirin E, Parsia B, Wu Dan, et al. HTN Planning for Web Service Composition Using SHOP2 [J]. Journal of Web Semantics, 2004(4): 377-396.
- [10] Narayanan, S, McIlraith S. Simulation, verification and automated composition of web services [C]//in: Proceedings of the Eleventh International World Wide Web Conference. Honolulu, Hawaii: [s. n.], 2002.
- [11] 方其庆, 彭晓明, 刘庆华, 等. 结合 AI 规划和工作流的动态服务组合框架研究 [J]. 计算机科学, 2009, 36(9): 110-114.
- [12] Johnson R, Hoeller J, Arendsen A, et al. Professional Java Development with the Spring Framework [M]. Beijing: China Machine Press, 2006: 233-238.

(上接第 52 页)

- [6] 代六玲, 黄河燕, 陈肇雄. 中文文本分类中特征抽取方法的比较研究 [J]. 中文信息学报, 2003, 18(1): 26-32.
- [7] 单松巍, 冯是聪, 李晓明. 几种典型特征选取方法在中文网页分类上的效果比较 [J]. 计算机工程与应用, 2003(22): 146-148.
- [8] 万忠, 张燕平, 张玲, 等. 基于覆盖算法决策界的特征选择算法 [J]. 计算机技术与发展, 2006, 16(4): 84-87.
- [9] Dunning T E. Accurate methods or the statistics of surprise and coincidence [J]. Computational Linguistics, 1993, 19(1):

61-71.

- [10] 闫屹, 张燕平, 耿媛. 基于 CHI 值特征选取和覆盖的文本分类方法 [J]. 计算机技术与发展, 2008, 18(5): 79-85.
- [11] 段震, 王倩倩, 张燕平, 等. 覆盖算法下文本分类特征选择的研究 [J]. 计算机技术与发展, 2008, 18(11): 29-31.
- [12] 张铃, 张钺. 佳点集遗传算法 [J]. 计算机学报, 2001, 24(9): 1-9.
- [13] 任江涛, 卓晓岚, 许盛灿, 等. 基于 PSO 面向 K 近邻分类的特征权重学习算法 [J]. 计算机科学, 2007, 34(5): 187-189.