

文章编号: 2095—0411 (2011) 01—0019—04

# 一种基于本体服务索引的 Web 服务扩展方法<sup>\*</sup>

徐守坤, 薛 浩, 李 宁, 马正华

(常州大学 信息科学与工程学院, 江苏 常州 213164)

**摘要:** 传统的获取 Web 服务的方法是通过关键字匹配, 由于此方式在发现潜在语义相符的 Web 服务时较为困难, 越来越多的学者研究如何基于语义检索 Web 服务。本文设计了一种新的语义匹配模式: 基于本体服务索引的服务匹配。通过建立从服务到本体库的索引, 将本体库和 Web 服务库关联成一个逻辑整体。通过 UDDI, 用户请求会直接发送给推理规则库, 在得到服务请求后, 通过相应推理规则得到领域本体中与服务请求在一定相似度范围内的类及实例, 而后将这个结果集输入到本体服务索引, 经过匹配可输出较符合的 Web 服务结果集。

**关键词:** 本体; 本体服务索引; Web 服务扩展

**中图分类号:** TP 393

**文献标识码:** A

## An Approach to Web Service Extension Based on Ontology Indexing of Service

XU Shou—kun, XUE Hao, LI Ning, MA Zheng—hua

(School of Information Science and Engineering, Changzhou University, Changzhou 213164, China)

**Abstract:** Keyword matching is a traditional method of web service. However, by the method it is very difficult to find out the potential semantics contained in web service. Therefore, more and more scholars are studying how to query the web service based on semantics. This paper proposes a new semantic matching mode based on the ontology service indexing. By building an indexing from service to ontology, the service and ontology can be regarded as a logic entity. The requests from users are sent to the rule base directly by UDDI, and then reasoning rules can be used to get classes and instances of ontology in a proper range. Finally, the result set is inputted into the ontology service indexing, with which the user could get a better web service result set via the matching.

**Key words:** ontology; ontology service indexing; web service extension

根据 W3C 的定义, Web 服务是一种通过 URI 标识的软件应用, 其接口及绑定形式可以通过 XML 标准定义、描述、检索, 并能通过 XML 消息及互联网协议完成与其他应用的直接交互<sup>[1]</sup>。随着 Web 服务技术的日益成熟, 许多机构将其核心业务作为一个 Web 服务放在 Internet 上, 以实现

更多的自动化和全球范围访问。如何准确高效地从庞大的 Web 服务群中找到所需服务, 并进一步满足自动化智能化的服务发现、执行、组合及互操作的要求, 是 Web 服务技术面临的巨大挑战。传统的 Web 信息检索是基于关键词的检索, 要想使机器在语义上理解 Web 上的资源, 必须使 Web 上的

<sup>\*</sup> 收稿日期: 2010—10—16

作者简介: 徐守坤 (1972—), 男, 吉林蛟河人, 博士, 副教授。

信息拥有明确定义的语义,而这种语义能够使计算机与人或其他计算机进行更好的合作,达到语义检索的目的。目前使用基于本体语义检索是比较活跃的研究方向,本文正是通过基于本体服务索引的方式来进行研究的。

## 1 本体

### 1.1 本体的概念

目前对本体的定义还没有统一的定论,但是对于本体的认识渐渐趋于成熟。其中最著名、被引用最为广泛的定义是由 Gruber 提出的:“本体是概念化的明确的规范说明”<sup>[2]</sup>,本体的概念包括四个主要方面:概念化、明确、形式化、共享。

### 1.2 旅游领域本体构建

领域本体通常采用迭代的方法进行构建,即通过不断的修改和完善来补充和修正本体<sup>[3]</sup>。

域本体的构建一般包含以下步骤(以旅游本体为例):①定义类层次结构。涉及到旅游相关的基本概念包括城市、景区、路线、酒店、票务、交通、天气等。每个概念都是一个层次结构,下层概念和上层概念是 IS—A 的关系。以景区为例,根据景区中旅游资源的性状特点,可分为人文地理和自然风光两个子概念。②定义本体类属性。本体类属性主要有对象属性(Objecttype)和数据属性(Datatype),对象属性主要是将个体关联到个体,而数据属性主要是将个体关联到数据类型<sup>[4]</sup>。③定义本体类属性的性质。主要有:传递属性、对称属性、函数型属性、逆属性、反函数属性。④创建本体实例,选择概念类并创建相应类的个体。

## 2 旅游领域推理规则设计

交通:旅行者可以根据自身需要,选择不同的方式到达景点,主要包括:公共交通、自己驾车、旅行社安排。

门票:根据不同人群、不同景点的打折情况以及旅行社与景区的合作情况,实行不同的票价。

住宿:不同人群根据自身的需要及经济情况安排相应的住宿。

满意度:每个旅行者可以根据交通、门票、住宿以及期间的天气、购物、娱乐等服务情况得到此次旅行的整体满意度。

对于以上问题将有不同的规则如下:

(1) 对于交通问题,根据不同的人和情况可能选择汽车、火车、飞机等。相应规则如下:

Rule1:旅行者所在地(? x) 景点所在地(? x, ? y) swrlb: lessThan(? y, 300km) → 交通类型(? x, 汽车)

Rule2:旅行者所在地(? x) 景点所在地(? x, ? y) swrlb: between(? y, 300km, 1 000km) → 交通类型(? x, 火车)

Rule3:旅行者所在地(? x) 景点所在地(? x, ? y) swrlb: between(? y, 国外) → 交通类型(? x, 飞机)

(2) 对于门票问题,根据不同的人不同的旅行社会享有不同的折扣价格:

Rule4:旅行者(? lxz) 旅行者证件(? lxz, ? certificate) 旅行社(? lxs) 景点(? jd, ? price) 景点活动(? jdhd, ? discount) → 门票(? ticket, ? price, ? certificate, ? discount)

(3) 对于住宿问题,根据不同的人不同的情况会享有不同的折扣:

Rule5:旅行者(? lxz) 旅行社(? lxs) 宾馆(? bg, ? price) 宾馆活动(? bghd, ? discount) → 价格(? lxz, ? ticket, ? discount)

(4) 对于满意度问题<sup>[7]</sup>,根据不同的人、交通方式、旅行社、宾馆住宿、城市及购物娱乐等等,都会影响满意度,规则如下:

Rule6:旅行者(? lxz) 旅游城市(? lycs) 天气(? tq, 晴) 风力(? tq, 2) 出行时间(? cysj, ? date) 温度(? tq, ? wd,) swrlb: between(? wd, 10, 24) 交通类型(? lxz, 汽车) 门票(? lxz, 恐龙园, ¥100) 住宿(? lxz, ? price) swrlb: lessThan(? price, 100) → 满意度(? myd, 高)

## 3 本体服务索引

本体服务索引 OIS (Ontology Indexing of Service) 的应用架构如下图 1 所示。首先通过 UDDI, 用户请求直接发送给推理规则库<sup>[5]</sup>, 经过一定的规则推理, 得到本体库中与服务请求在一定相似度范围内的类及实例, 而后将这个结果集输入到本体服务索引, 服务索引在得到类和实例之后, 通过算法可得到符合条件的服务地址, 最后服务库将所有的目标服务的地址通过 UDDI 输出给用户。

### 3.1 本体服务索引的结构

本体服务索引结构主要包括 3 个部分: 服务引

用地址 (WSUri)、索引模式 (Model)、服务参数 (Params)。WSUri: 是对应的 Web 服务的引用地址, 此地址包括服务的名称; Model: 对应于本体的中的基本元素名称, 如类名称、实例名称等; Params: 服务参数, 包括输入和输出参数, 结构见图 2。

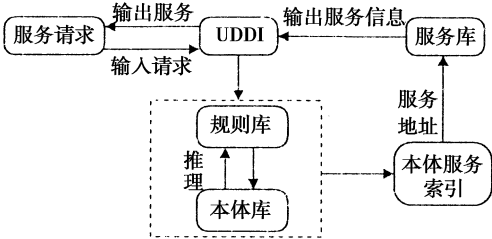


图 1 本体服务索引的应用架构  
Fig. 1 Application framework of OIS

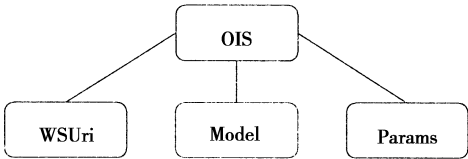


图 2 本体服务索引的组成结构  
Fig. 2 Composition and structure of OIS

3.2 本体服务索引的数据组成

按照 3.1 中定义的结构, 建立本体服务索引需要 3 个方面的数据。①服务引用地址: 服务注册者在 UDDI 注册时提供的引用地址; ②索引模式: 服务提供者在注册服务时, 需要同时根据服务的属性、接口、功能, 从对应的领域本体中选择相应的类名或实例名, 便可作为索引模式; ③服务参数: 包括输入和输出参数, 见图 3。

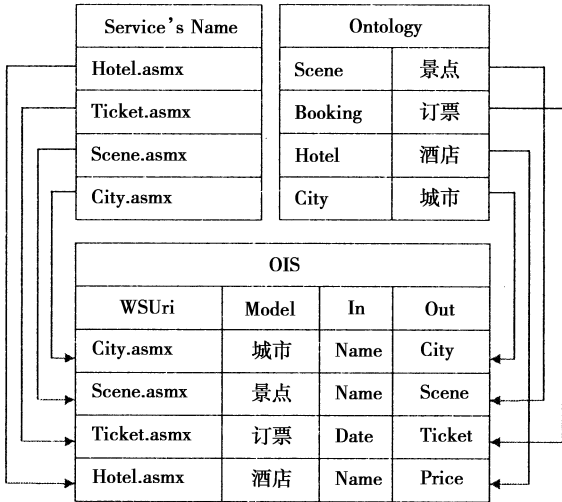


图 3 本体服务索引的数据组成  
Fig. 3 Data composition of OIS

服务的本体索引 (OIS) 的数据主要来源于 Ontology 和 Web 服务注册信息, 通过组合这些信息即可建立 OIS。而这些信息的融合主要靠服务提供者来操作完成, 当服务提供者在 UDDI 注册自己的 Web 服务时, 服务提供者不但要提供服务的基本信息, 还要考虑此服务与对应本体的关联, 即属于哪个本体, 以及对应这个本体中的那个子类或实例。

3.3 基于本体服务索引的 Web 服务查询

下面将以一个简单的旅游本体为例来说明如何运用本体服务索引, 见图 4。

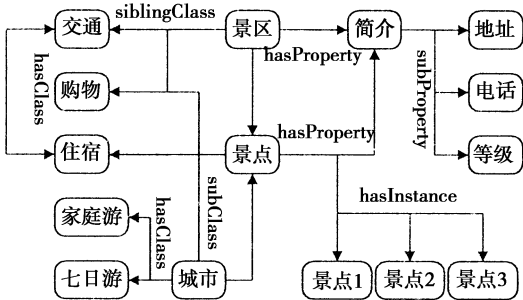


图 4 旅游本体的局部视图  
Fig. 4 Local view of travel ontology

(1) 本体语义的规则推理算法。

算法 1: 本体规则推理算法

输入: 用户请求 (User Request)

输出: 本体推理结果集 (OntoResults)

```
Onto_Rule (string [] UserRequest) {  
    List<string> RuleItem = Rule. Base; // 规则库  
    List<string> OntoItem = Onto. Base; // 本体库  
    List<string> OntoResults; // 本体结果集  
    for (int i=0; i< UserRequest. Length; i++) {  
        // 根据输入条件和规则获取结果  
        if (LHS (UserRequestm, RuleItem) == true)  
            OntoResults. Add (OntoItem [i]);  
    }  
    return OntoResults; // 返回结果  
}
```

(2) 通过本体推理将得到的推理结果输入本体服务索引, 查询得到相关的服务, 返回引用服务需要的信息, 如引用地址、输入参数和输出参数等。

算法如下:

算法 2: 本体服务索引查询算法

输入: 本体请求的推理结果集 (OntoResults)

输出: 服务的信息集 (ServResults)

```
OSI_Query (string [] OntoResults) {
    // 获取索引中的 Model
    List<string> servModel= OIS. Model;
    // 获取索引中的 Uri
    List<string> servUri= OIS. Uri;
    // 获取索引中的 ParamIn
    List<string> servParamIn= OIS. ParamIn;
    // 保存服务的参数地址
    List<string> servDestUri, servDestParamIn;
    for (int i=0; i< OntoResults. Length; i++) {
        for (int j=0; j<servUri. Count; j++)
            if (servUri [j] ==OntoResults [i]) {
                servDestUri. Add (servUri [j]);
                servDestParamIn. Add (servParamIn [j]);
            }
    }
    return servDestUri; // 返回结果
}
```

(3) 两种查询方法的实例比较。

传统的对于 Web 服务的查询主要是基于关键字匹配<sup>[6]</sup>, 这种查询方法比较简单, 且易于实现, 所以在查询中得到了普遍的运用; 而本文提出的基于本体服务索引查询的方法是基于语义本体的查询方法, 此种方法可以很好地利用本体的语义特性, 查找潜在的目标服务。查询实例如下: 查询请求为北京七日游。方法 1: 关键字匹配; 查询过程: 通过简单的分词, 查找服务中带有“北京”或“七日游”关键字的服务; 查询结果: ①城市 (北京) 相关的 Web 服务集, ②“七日游”相关的 Web 服务集。方法 2: 基于本体服务索引的查询; 查询过程: 利用适当的分词方法, 得到关键字“北京”和“七日游”, 由图 4 的旅游本体可知, “七日游”属于路线类, 而路线又与景点、交通、住宿、购物等相关, 这样就可以通过它们的关系得到相关的所有服务集; 查询结果: ①城市 (北京) 和“七日游”

相关的 Web 服务集, ②交通、娱乐、住宿等服务集③参观的景点服务集。

由实例可以看出, 利用“关键字”匹配的方法仅能查询到基本服务, 而通过本体服务索引的查询方法, 不但能得到基本服务, 同时能得到相关的拓展服务。而且, 如果服务库中没有对应关键字的服务, 但有与关键字相关的潜在服务, 前者不能得到这些潜在服务, 而通过规则库的推理, 可以得到潜在服务的对应的实例或类, 然后结合服务索引, 便可提取到潜在服务。

## 4 结 语

本文简述了领域本体的创建过程, 并设计了创建旅游领域的系列规则, 在此基础上引入了一种新的服务检索方法, 即本体服务索引, 且借助领域本体的精确定义与可推理优势, 可以实现 Web 服务的检索。通过本文建立的本体到 Web 服务的本体服务索引便可得到目标 Web 服务, 与传统的基于关键字的服务检索的方式相比, 本方法在输出服务的相关性有了一定的改进。利用本体的语义相关性和规则的推理性, 今后将是在此基础上研究如何更快更好地发现目标服务, 即在现有的规则和服务索引基础上如何自动 (半自动) 进行本体推理和建立服务索引。

## 参考文献:

- [1] W3C. Web Services Glossary [EB/OL]. [2004-02-11]. <http://www.w3.org/TR/ws-gloss>.
- [2] 陆建江, 张亚非, 苗壮, 等. 语义网原理与技术 [M]. 北京: 科学出版社, 2007: 66-67.
- [3] 胡艳丽, 张维明, 肖卫东, 等. 内置谓词函数依赖及其推理规则 [J]. 国防科技大学学报, 2009, 31 (5): 58-63.
- [4] 孙明, 陈波, 周明天. 结合 DLsafe 规则发现日志本体频繁模式的方法 [J]. 电子学报, 2010, 2 (2): 376-381.
- [5] 张维明, 宋峻峰. 面向语义 Web 的领域本体表示、推理与集成研究 [J]. 计算机研究与发展, 2006, 43 (1): 101-108.
- [6] 华进, 钱雪忠, 李长江. 基于语义的 Web 服务发现模型研究 [J]. 计算机工程与设计, 2008, 29 (9): 2 394-2 396.
- [7] 许德山, 乔晓东, 朱礼军, 等. 本体推理在知识检索中的应用 [J]. 现代图书情报技术, 2009 (1): 58-63.