

# 基于本体的 Web 服务发现模型研究

郭学俊, 杜 春, 秦雪杰

(河海大学 计算机及信息工程学院, 江苏 南京 210098)

**摘 要:**针对传统的 Web 服务发现机制查准率低的缺点, 结合本体技术与 OWL-S 语义描述语言设计一个 Web 服务发现模型。该模型主要包括查询处理器、领域本体库和发现引擎三个模块, 通过在领域内共同的语义理解基础上抽取服务查询信息中所需的功能信息以及对服务广告信息的语义描述, 实现针对服务功能信息的语义匹配, 从而提高服务的查准率。

**关键词:**本体; OWL-S; Web 服务发现

**中图分类号:** TP18

**文献标识码:** A

**文章编号:** 1673-629X(2006)06-0207-03

## Research of an Ontology Based Web Services Discovery Model

GUO Xue-jun, DU Chun, QIN Xue-jie

(Computer and Information Engineering College, Hohai University, Nanjing 210098, China)

**Abstract:** Aimed at the pitfall of low precision existing caused by mechanism of Web services discovery, this paper puts forward a Web services discovery model based on ontology. This model mainly includes query processor, domain ontology DB, and discovery engine components. At the basis of common semantic understanding, the model extracts the function information from both Web services query information and Web services advertisement description, and implements semantic based matching according to the Web services capability information, so as to improve the precision of Web services discovery.

**Key words:** ontology; OWL-S; Web services discovery

### 0 引 言

Web 服务发现是 Web 服务系统架构的重要部分, 发现符合用户需求的服务是实现服务复用、组合的重要前提, 服务发现的效果直接关系到服务调用的质量, 影响到服务组合的相容性和可替换性, 关系到能否真正实现服务的即插即用。在服务请求者—服务提供者—服务注册中心的 Web 服务架构中, 服务请求者向服务注册中心发出查询请求时, 服务注册中心将请求映射为服务请求描述并与广告描述进行匹配, 服务发现的问题就转化为请求描述与广告描述之间的匹配问题。然而由于采用语法级的服务描述语言和基于关键字的匹配算法, 传统的 Web 服务发现机制存在查准率低的不足。Web 服务采用语法级别的 WSDL 描述语言进行描述, 着重描述服务接口的语法而缺乏对服务功能的语义描述。Web 服务的匹配是通过在 UDDI 上注册过的服务名称、服务标识、服务属性等信息进行关键字匹配来实现的, 缺乏对用户查询请求中所需服务能力信息的提取和处理, 同时服务请求者无法从广告描述中获取服务功能信息。因此其查准率不高, 很难保证服务调用的质量, 影响服务自动组合的效果。

针对上述 Web 服务发现机制的缺陷, 业界已提出不

少解决办法。Mark Klein 等提出了一种基于过程本体的 Web 服务发现方法, 利用过程本体来描述查询请求与 Web 服务, 服务的发现是通过把查询的过程本体与服务的过程本体相匹配得到的<sup>[1]</sup>。Paolucci Massimo 等提出了基于 DAML-S 的 augment UDDI 注册系统, 使得 UDDI 注册中心中增加了服务的语义信息<sup>[2]</sup>。乔治亚大学的 SpeedR 项目提出了在 P2P 环境下的服务发现方法<sup>[3]</sup>。

综观现有研究, 其各自不同的服务发现方法都考虑利用了语义信息和本体论, 因此文中进一步给出限定于特定领域的服务发现模型。在该模型中引入本体论方法解决 Web 服务发现过程中存在的语义异构问题, 达成领域内一致的语义理解。采用语义 Web 服务本体描述语言 (OWL-S) 对 Web 服务进行语义描述, 使用户能够了解服务的功能细节, 加强服务描述信息的机器可理解性, 支持用户需求与服务能力之间的逻辑推理匹配。同时在该模型中采用基于语义的匹配算法, 并结合逻辑推理进行匹配以提高服务查准率。

### 1 基于本体的 Web 服务发现模型设计

要实现基于功能的服务发现, 首先必须解决因缺乏共同语义解释而造成的语义冲突问题, 其次应考虑添加 Web 服务的语义信息使用户获得服务功能信息, 因此需要采用本体论建立领域本体库, 为领域信息提供统一的描述和表现形式。同时利用 OWL-S 对 Web 服务进行语义

收稿日期: 2005-09-01

作者简介: 郭学俊 (1954-), 男, 江苏南京人, 副教授, 研究方向为计算机网络与分布式处理。

描述,使用户能够了解服务的功能细节,加强服务描述信息的机器可理解性。Web 服务发现的实质是请求描述与广告描述之间的匹配问题,而采用关键字匹配的方式缺乏对用户查询请求中所需信息的提取和处理,不能很好地描述用户查询请求中所需服务的能力(capability),并且服务请求者无法从广告描述中获悉服务功能信息,因此需要构造查询处理器来抽取查询请求中的服务特征信息,形成标准的语义请求描述,同时需要建立发现引擎实现针对服务功能信息的语义匹配,从而提高服务查准率。基于本体的 Web 服务发现模型如图 1 所示。

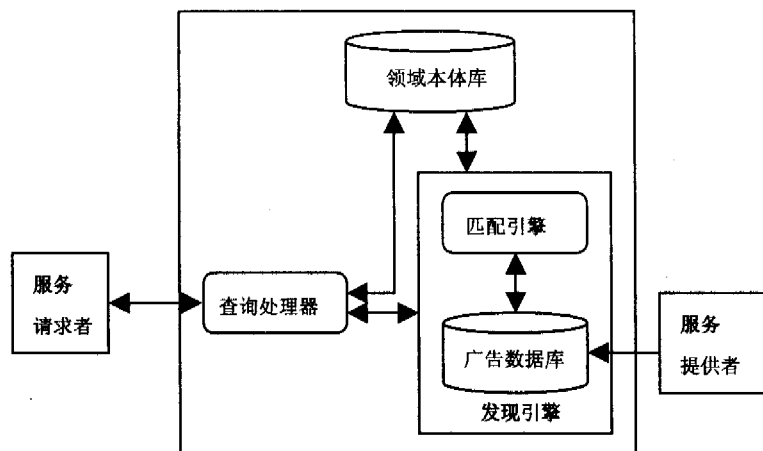


图 1 Web 服务发现模型结构

※ 查询处理器:接受用户输入的查询请求,根据本体库对用户的查询信息进行标准化和过滤,保留应用于查找的条件和约束信息,形成新的用户查询描述。该模块能够从用户的查询请求中抽取出满足其需求的服务功能信息,并形成标准、无语义冲突的查询描述。

※ 领域本体库:存储特定领域内的知识库、知识关系。建立特定的领域本体,使本领域内的用户形成共同的语义理解。

※ 发现引擎:该模块包含匹配引擎与广告数据库两个组件。匹配引擎利用基于语义的匹配算法对由查询处理器产生的查询描述与服务提供者广告描述进行匹配,并结合建立在本体上的推理系统进行逻辑推理。广告数据库用来存储服务提供者的服务描述信息,该服务描述信息是由 OWL-S 描述语言所描述的。该发现引擎建立在语义基础上的匹配,因此能大大提高服务的查准率。

## 2 主要模块功能设计与实现

结合旅行在线预订领域对本发现模型进行设计与实现,并以其中的旅馆预订服务为例对以下 3 个主要模块具体加以说明。

### 2.1 领域本体库建立

本体论(Ontology)是用于描述某一领域知识正式的、精确的形式化规范<sup>[4]</sup>,是在语义和知识层次上描述信息的概念模型和建模工具。文中采用标准 Web 本体语义(OWL)来建立领域本体库,领域本体库存储领域内服务

本体信息,包含服务的概念以及服务间的关系。构建本体库包含 3 个方面内容:a. 服务的概念(名称),对应于本体论的 class;b. 服务描述所涉及的属性概念,对应于本体论的 attribute;c. 同时也定义了概念之间的关系,如相同、包含、实例等关系。

本模型应用于旅行在线预订领域,因此首先需建立其领域本体库。图 2 是旅行服务预订的本体。该领域本体包含旅行(Trip)、交通(Transportation)、旅馆(Hotel)预订 3 个本体类。交通本体和旅馆本体是旅行本体的子类,旅行本体定义数据类型属性 TripStartDate 和 TripEndDate,分别对应旅行起始时间和结束时间。交通本体(Transportation Ontology)包含了离开时间、离开地点、到达时间和地点的信息,同时定义了轿车(car)、火车(train)、飞机(flight)子类。旅馆本体(Hotel Ontology)定义了旅馆名字、旅馆地址、旅馆电话、旅馆房间等属性,旅馆房间(HotelRoom)类是旅馆类的子类,包含房间类型、价格等信息。

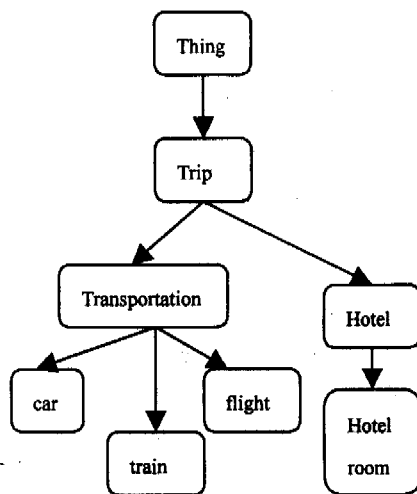


图 2 Trip Ontology

旅馆预订服务本体(Hotel Ontology)部分代码如下:

```
<owl:Class rdf:ID="HotelRoom">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Hotel">
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="# Hotel">
  <rdfs:subClassOf rdf:resource="# Trip">
...
<owl:DatatypeProperty rdf:ID="HotelName">
<owl:domain rdf:resource="# Hotel"/>
<owl:range rdf:resource="xsd:string"/>
</owl:DatatypeProperty>
...
```

### 2.2 Web 服务查询处理器

该查询处理器模块的作用是提取用户查询请求中有用的服务能力信息并进行适当推理,使得经过映射后的服

务请求描述更能精确表达用户所需的服务信息。文中利用 OWL-S 的 Service Profile 中的 IOPE 信息对请求条件与服务描述进行匹配,因此对于用户请求的查询信息需提供服务功能描述信息、输入参数、输出参数等信息。该查询处理模型执行步骤如下:

(1) 接受用户输入查询条件,包括服务描述信息、输入参数、输出参数等;

(2) 参照领域本体库,把不存在于领域本体中的查询条件或参数去除;

(3) 把接收到的查询信息进行标准化,例如标准词与别名之间的映射等,以达到领域内统一的表现形式;

(4) 把查询信息转化成 OWL-S 的 Service Profile 来描述;

(5) 调用发现引擎进行匹配。

以查询旅馆服务为例,假设用户提出一个查询请求:旅馆所在地为北京、房间容纳少于 3 人、价格低于 600RMB。文中把 Service Profile 作为公共的超类(super-class),服务请求描述与广告描述都是其子类,利用描述逻辑来代替 OWL 语法进行描述,那么该查询请求通过查询处理器可以最终描述为:

$$\text{QueryProfile} \equiv \text{ServiceProfile} \cap (\exists \text{HotelLocation. Beijing}) \cap (\leq \text{RoomCapacity. 2}) \cap (\leq \text{RoomPrice. 600})$$

### 2.3 Web 服务发现引擎

该模块分为两部分:一是匹配引擎,其接收由查询处理器输出的服务请求描述,并查询广告数据库中的所有广告,利用基于语义的匹配算法对服务描述进行匹配,该匹配引擎是整个模型的重要部分;二是广告数据库,用来存放服务的广告信息,广告数据库存放利用 OWL-S 描述的服务信息,增加对服务功能的语义描述。

匹配推理过程主要是利用服务描述中的能力描述和用户需求描述中的目标描述,用一阶谓词逻辑推理机判断该服务是否能满足的用户要求。假设  $a$  是注册中心所有广告的集合,  $Q$  为用户需求中的目标描述,匹配算法返回一致(compatible)的广告集,匹配的形式化定义为:

$$\text{matches}(Q) = \{A \in a \mid \text{compatible}(A, Q)\}$$

服务能力描述  $D1$  与用户需求的目标描述  $D2$  是一致的(compatible),只有当它们的交集(intersection)是可满足的(satisfiable):

$$\text{compatible}(D1, D2) \Leftrightarrow \neg (D1 \cap D2 \subseteq \perp)$$

按照匹配精确程度的差异性,可定义以下几个匹配类型<sup>[5]</sup>:精确匹配(Exact)、可替代匹配(PlugIn)、包含匹配(Subsume)、相交匹配(Intersection)、失败匹配(Disjoint)。匹配精确度由精确匹配类型到匹配失败类型是逐渐降低的。精确匹配是指服务广告  $A$  与用户请求  $R$  是两个相同的概念,则为  $\text{Exact}(R, A)$ ,即:  $A = R$ 。若用户请求  $R$  是服务广告  $A$  的子概念,则为可替代匹配类型,表示为  $\text{PlugIn}(R, A)$ ,即  $R \subseteq A$ 。当用户请求  $R$  是服务广告  $A$  的超级概念,则为可代替匹配类型,表示为  $\text{Subsume}(R, A)$ ,即  $A \subseteq$

$R$ 。相交匹配是指若用户请求  $R$  与服务广告  $A$  的交集是可满足的,则为  $\text{Intersection}(R, A)$ ,即  $\neg (A \cap R \subseteq \perp)$ 。其余的情形视为匹配失败,即  $A \cap R \subseteq \perp$ 。

本模型采用文献[6]提出的基于语义的匹配算法对服务广告与用户请求进行匹配,用户请求通过一个主循环与所有的广告进行匹配,根据上述的匹配类型来计算各广告的匹配程度并把匹配结果进行排列,最后返回匹配程度最高的服务广告描述。具体代码如下:

```
match(request) {
    recordMatch = empty list
    for all adv in advertisements do
        if match(request, adv) then
            recordMatch.append(request, adv)
    return sort(recordMatch); }
```

服务广告与用户请求的匹配包括请求的输出与广告的输出之间的匹配和广告的输入与请求的输入之间的匹配这两方面,输入匹配算法与输出匹配算法类似。输出匹配算法的部分代码如下:

```
outputMatch(outputRequest, outputsAdvertisement) {
    globalDegreeMatch = Exact
    for all outR in outputRequest do
        find outA in outputsAdvertisement such that
            degreeMatch = maxDegreeMatch(outR, outA)
        if (degreeMatch = fail) return fail
        if (degreeMatch < globalDegreeMatch)
            globalDegreeMatch = degreeMatch
    return sort(recordMatch); }
```

### 3 试验与分析

为了提高 Web 服务发现的查准率,本中给出一个基于本体的服务发现模型,并结合旅行预订领域搭建该 Web 服务发现模型试验平台。本实验在 J2EE1.4 平台下开发、部署 Web 服务;在 protégé 3.0 建立领域本体库及对 Web 服务进行语义描述,并将 Racer 推理机作为 protégé 的插件进行推理。把查准率作为度量 Web 服务发现性能的主要指标,采集多个旅行在线领域内的 Web 服务作为样本集,通过仿真性能测试比较传统的关键字匹配方法和本服务发现模型。试验结果表明在执行效率上,本服务发现模型利用语义分析作为查询的预处理,可以有效地减少检索目标服务的数量,提高查询的效率;在服务发现性能方面,由于本模型从服务信息的发布到服务的查找都利用语义信息来约束服务,因此同传统的关键字匹配方法比较可以获得更准确的语义信息,进而能更精确地定位服务,提高查准率并改善 Web 服务发现性能。

### 4 结束语

设计基于本体的 Web 服务发现模型,该模型不仅能提高服务查找的效率,而且较传统的关键字匹配方法能更

(下转第 212 页)

Queue Interface, MQI)、通道(Channel)。应用程序使用 MQ 的 MQI 函数和队列管理器(MQ 运行时刻的程序)进行通信。队列管理器在工作时,它需要用到对象如队列和通道。同时,队列管理器也是一个对象。

以下为电信 IP 电话业务中预付费和后付费结转具体接口的多层分布体系事例。MQ 消息中间件在 IP 电话业务系统的通道设置如图 4 所示。

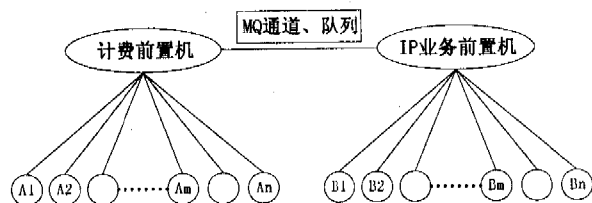


图 4 MQ 在 IP 电话业务系统的通道设置

#### (1) IP 电话业务系统中计费通道设置。

- 端口号设置(etc/services);
- 创建队列和通道(包括发送通道和接收通道);
- 运行侦听程序;
- 启动发送通道:在发送方,使用 amqsput 程序通过 MQI 发送一条信息至接收方;在接收方,使用 amqsget 程序从队列获取消息。

#### (2) IP 电话业务系统中 IP 通道设置。

设置过程为:创建队列管理器→启动队列管理器→进入命令模式→创建传输队列→创建本地队列→创建远程队列→创建通道→运行侦听程序→启动发送通道

在通道建立并侦听后即可通过相关函数进行消息接收和发送。MQ 消息接收的简易流程如图 5 所示。

### 3 结束语

随着电信 IP 增值业务不断发展,客户化需求多样化,电信计费账务的省集中、全国集中模式的发展趋势,以及

随着基于 Web 的瘦客户机结构的发展,基于多层分布体系的应用将会越来越广泛。而中间件作为分布体系应用的关键技术,以其独特的优势为各种分布式应用的开发注入了强大动力,极大地推动了 IP 业务系统及其它应用系统集成的发展。

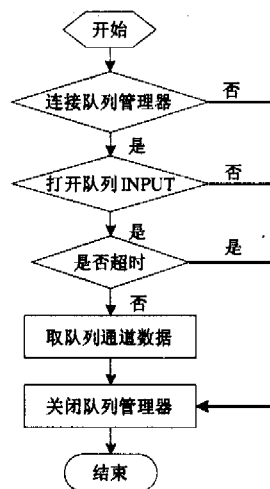


图 5 MQ 消息接收流程

#### 参考文献:

- [1] Coulouris G. 分布式系统概念与设计[M]. 北京:机械工业出版社,2004.
- [2] 李志彤,闫艳生. 多层分布式体系技术及其应用[J]. 计算机工程与应用,2003(25):95-97.
- [3] 成奋华,刘长青. 中间件与多层分布式体系技术[J]. 企业技术开发,2003(6):24-25.
- [4] 罗会兰,谢小兵. 多层分布式体系技术在电信触摸屏多媒体查询系统中的应用[J]. 微型机与应用,2004(1):36-38.
- [5] 周劲,谷岩. 基于中间件技术的多层分布式应用系统的开发[J]. 微机发展,2003,13(8):33-35.

(上接第 209 页)

精确地定位服务,从而提高查准率,改善 Web 服务发现性能。对如何利用本体论和推理机制来改进 Web 服务发现机制,实现 Web 服务最佳匹配的领域进行一定的研究与试验。下一步的工作重点是在服务匹配中添加前置条件与约束信息进一步提高查准率。

#### 参考文献:

- [1] Klein M, Bernstein A. Searching for Services on the Semantic Web Using Process Ontologies[A]. In: Proceedings of the International Semantic Web Working Symposium(SWWS)[C]. Amsterdam: IOS Press, 2001. 159-172.
- [2] Massimo P, Takahiro K, Payne T R, et al. Importing the semantic Web in UDDI[A]. In: Proceedings of Web Services, E-business and Semantic Web Workshop (CAiSE Workshop)[C]. Toronto, Canada: [s. n.], 2002. 225-236.
- [3] Sivashanmugam K, Verma K, Mulye R, et al. SpeedR: Se-

mantic P2P Environment for diverse Web services registries [Z]. Final Presentation, CSC1:8350, Enterprise Integration, CA: Department of Computer Science, University of Georgia, 2002.

- [4] Stumme G. Using ontologies and formal concept analysis for organizing business knowledge[A]. Becker J, Knackstedt R. Wissensmanagement mit Referenzmodelloen - Konzepte für die Anwendungssystem - und Organisationsgestaltung, Physica [C]. Heidelberg: [s. n.], 2002. 163-174.
- [5] Li Lei, Horrocks I. A software framework for matchmaking based on semantic web technology[M]. New York, USA: ACM Press, 2003. 331-339.
- [6] Massimo P, Soudry J, Srinivasan N, et al. A Broker for OWL - S Web Services[A]. In: Proceedings of the AAAI Spring Symposium[C]. Palo Alto, California: [s. n.], 2004. 92-99.