

# 基于普适计算的服务发现系统设计

姚梦雪, 万晓冬

(南京航空航天大学 自动化学院, 江苏 南京 210016)

**摘要:**在普适计算环境下,服务是动态变化的,而服务发现技术能够在动态变化的普适计算环境中动态查找、定位所需要的服务,是解决普适计算环境异构性、实现不同应用集成的有效途径。介绍了服务的语义描述语言 OWL - S,用于对服务进行描述,然后设计并实现了一个普适计算下的服务发现系统。同时,详细介绍了服务请求时进行服务匹配使用的匹配算法。通过增加语义信息,达到服务发布者和请求者可以实现语义层次的匹配,提高了服务的查准率和查全率。

**关键词:**普适计算;服务发现;OWL - S;本体;服务匹配

**中图分类号:**TP311.5

**文献标识码:**A

**文章编号:**1673 - 629X(2009)03 - 0130 - 04

## Design of Service Discovery System Based on Pervasive Computing

YAO Meng-xue, WAN Xiao-dong

(College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

**Abstract:** In the pervasive computing environment, the service is changing dynamically, and the service discovery technology can lookup and locate the necessary services in the dynamic changing pervasive computing environment. This is the effective way to solve the heterogeneous of the pervasive computing environment, and implement different application integration. In this paper, introduce the OWL - S, then design and implement a service discovery system in pervasive computing environment. At the time, introduce detailedly the service matching arithmetic.

**Key words:** pervasive computing; service discovery; OWL - S; ontology; service matching

### 0 引言

普适计算是信息空间与物理空间的融合,在这个融合的空间中人们可以随时随地和透明地获得数字化的服务;计算机本身将从人们的视线中消失,人们注意力的中心可以回归到要完成的任务本身<sup>[1]</sup>。

在普适计算中,服务发现的研究起着关键性的作用。服务发现包含了两个过程:一个是服务的发布;一个是服务的查询。服务提供者将服务的描述发布在服务注册中心;服务请求者则通过提交查询来得到所需服务的信息。

### 1 相关工作

语义 Web 的研究普遍采用本体论,语义 Web 中的本体表示的是人们对特定领域中的概念统一的、本质的认识。OWL(Web Ontology Language)<sup>[2]</sup>是 W3C 推

荐的本体描述语言的标准,它可以定义类、子类、等价类,并且可以定义属性和子属性,以及它们的约束,如:定义域(Domain)和值域(Range)等。文中用 OWL 来构建领域本体。

OWL - S<sup>[3]</sup>是基于 OWL 语言的 Web 服务本体,是 W3C 推荐的标准。OWL - S 定义了一组核心的语言构件,用于对 Web 服务进行逻辑化描述,所生成的描述文件支持机器理解,从而支持代理程序基于逻辑语义实现对 Web 服务的自动发现、调用、组合与监控。OWL - S 由三部分组成,如图 1 所示。

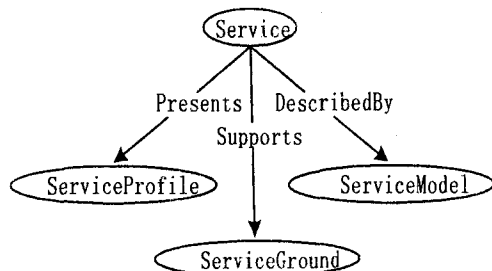


图 1 OWL - S 的顶层本体

(1) 服务概貌(ServiceProfile)。

ServiceProfile 是用来描述服务是做什么的。Servi-

收稿日期:2008 - 07 - 01

基金项目:武器装备预研基金项目(9140A04030307HK0201)

作者简介:姚梦雪(1983 - ),女,河南邓州人,硕士研究生,研究方向为普适计算下的服务发现;万晓冬,硕士,副研究员,研究方向为分布式仿真。

ceProfile 主要包括三类信息: ①服务提供者的信息, 如服务提供者物理地址、email 地址、电话等联系信息; ②服务的功能信息, 如服务的输入(Input)、输出(Output)、前提(Precondition)和效果(Effect); ③服务的非功能信息, 如服务质量 QoS、分类等。

### (2) 服务模型(ServiceModel)。

ServiceModel 是用来描述服务如何工作。ServiceModel 主要包括过程(Process)和过程控制(Process Control)两部分信息。其中, Process 以流程的形式描述服务中各个子活动的控制和协同关系。Process Control 管理和监控服务的运行。

### (3) 服务绑定(ServiceGrounding)。

ServiceGrounding 是用来描述如何访问服务。ServiceGrounding 描述访问服务的具体细节, 例如服务 URL 地址、传输协议、消息格式等。基于该本体提供的信息可实现从抽象描述到具体服务的映射。OWL-S 主要考虑了 Grounding 到 WSDL 的映射。

文中将 ServiceProfile 作为广告服务发布在服务注册中心中, 并利用 ServiceProfile 的信息进行服务匹配。同时, 服务请求也可将 ServiceProfile 作为表达服务查询条件的语言, 从而使得服务匹配能够更加方便。

## 2 系统的设计

服务发现系统结构如图 2 所示。

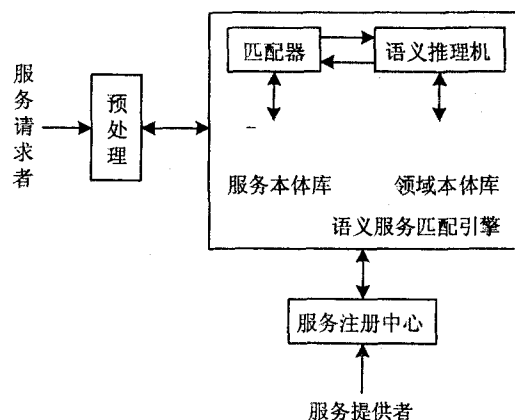


图2 服务发现系统结构图

图3是服务提供者发布服务时的执行流程图: 服务提供者首先用 OWL-S 对服务进行描述, 然后通过客户端提交, 存储到数据库中, 如果成功注册则返回服务的唯一标识符, 否则返回出错信息。

图4是服务请求者查找服务时的执行流程图: 服务请求者首先对服务进行描述, 然后在客户端提交。匹配器通过比较请求服务和数据库中已注册的服务, 返回匹配结果, 请求者可以匹配结果中选取匹配度最好的服务进行使用。

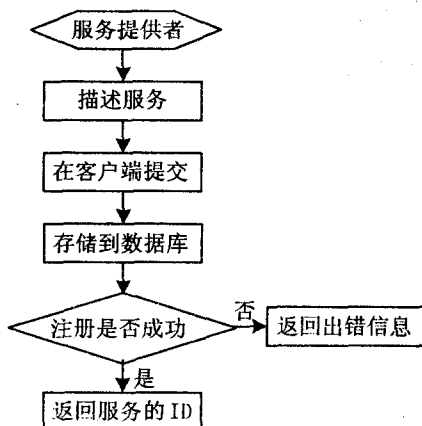


图3 服务提供者发布服务流程图

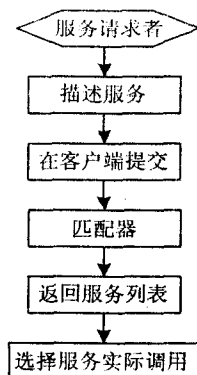


图4 服务请求者请求服务流程图

## 3 关键技术和算法

### 3.1 存储发布服务的 OWL-S 信息到数据库

通过 OWL-S 编辑器生成发布服务完整的 OWL-S 描述, 在客户端提交后, 系统采用数据库进行存储<sup>[4]</sup>。首先利用卡耐基-梅隆大学开发的 OWL-S API 对 Profile 文件进行解析, 以得到 Profile 文件中各部分的具体信息, 如服务提供者的联系信息、服务的输入/输出参数信息以及服务的性能参数等。

在把 OWL-S 描述的语义信息映射到数据库时, 使用推理机对本体进行推理。系统采用开源的 OWL-JessKB<sup>[5]</sup>进行本体的推理。OWLJessKB 是针对 OWL 语言而设计的描述逻辑推理机。OWLJessKB 提供 Java API 以支持读取 OWL 文件并对 OWL 文件进行信息检索。OWLJessKB 底层设计采用 Jena 和 Jess。OWLJessKB 支持基本的推理任务如包含关系、等价关系等的推理, 因而系统可以利用它来推理出 OWL-S 所描述的语义信息。

### 3.2 服务匹配算法

服务匹配是服务发现的一个重要部分, 服务匹配就是根据用户对服务接口的描述去寻找能够满足要求的服务。

文中从服务分类(ServiceCategory)、服务输入输出和服务质量(QoS)几个方面对广告服务和服务请求进行匹配。

文中的匹配算法是以函数 ConceptMatch() 为基础的, 函数 ConceptMatch() 的定义如下:

- a. 完全匹配(exact): 当请求概念与提供概念相等时;
- b. 插拔匹配(plugin): 当提供概念包含请求概念时;
- c. 包含匹配(subsume): 当请求概念包含提供概念时;
- d. 匹配失败(fail): 如果不属于上面三种情况则匹配失败。

匹配算法分三层进行匹配。

(1) 分类(ServiceCategory)匹配。

广告服务与请求服务的分类匹配分两种情况考虑:

- ① 二者引用的是同一本体;
- ② 二者引用的是不同的本体。

当二者引用的是不同的本体时, 直接过滤掉该候选服务; 当二者引用的是同一本体时, 若 ConceptMatch(Adv. ServiceCategory, Req. ServiceCategory) 为 Fail 时, 过滤掉该候选服务, 否则, 进行下一层次的匹配, 即功能性匹配。

(2) 功能性匹配<sup>[6]</sup>。

如有多个广告服务 ADVs 分别与一个请求服务匹配时, 用函数 CapabilityDegreeOfMatch() 来评价每一个广告服务 Adv<sub>i</sub> 与请求服务 Req 之间的匹配。函数 CapabilityDegreeOfMatch() 是以函数 ConceptDegree Of-Match() 为基础的。

ConceptDegreeOfMatch() 的定义如下:

ConceptDegreeOfMatch( $c_1, c_2$ ) =

$$\begin{cases} 0 & (\text{若 ConceptMatch}(c_1, c_2) = \text{exact}) \\ c_1 \text{ 和 } c_2 \text{ 之间的层数} & (\text{若 ConceptMatch}(c_1, c_2) = \text{plugin} \mid \text{subsume}) \\ \text{NULL} & (\text{若 ConceptMatch}(c_1, c_2) = \text{fail}) \end{cases}$$

然后用下面的式子给出函数 Capability Degree OfMatch() 的定义:

$$\begin{aligned} \text{CapabilityDegreeOfMatch}(\text{Adv}, \text{Req}) = & W_p \left( \left( \sum_{i=1}^{n_1} \text{ConceptDegreeOfMatch}(c_i, c'_i) \right) \right. \\ & (\text{当 ConceptMatch}(c_i, c'_i) = \text{plugin})) + \\ & W_s \left( \left( \sum_{i=1}^{n_2} \text{ConceptDegreeOfMatch}(c_i, c'_i) \right) \right. \\ & (\text{当 ConceptMatch}(c_i, c'_i) = \text{subsume})) \end{aligned}$$

式中,  $n_1$  和  $n_2$  是广告服务和请求服务功能中插拔匹配和包含匹配概念的个数,  $W_p$  和  $W_s$  是这两层匹配的权重, 因为定义完全匹配层的 ConceptDegreeOfMatch() 等于 0, 所以它没有被指定一个权重。

CapabilityDegreeOfMatch(Adv, Req) 最低的服务是首选的。

(3) 非功能性匹配。

如果两个或者两个以上广告服务的 Capability-DegreeOfMatch() 相同, 要在这些广告服务中选出最优的, 就要用到非功能性匹配, 在文中, 主要用到服务质量<sup>[7]</sup>的匹配。服务质量的匹配主要从 QoS 的三个属性进行比较, time, cost, reliability 分别表示服务响应时间、服务成本、服务可靠性。服务质量模型定义为  $QoS = (\text{time}, \text{cost}, \text{reliability})$ 。

质量参数的度量可以分为两类: 一类是质量度量的值越大代表该质量参数越优, 如可靠性, 另一类是质量度量的值越小代表该质量参数越优, 如响应时间, 服务成本。因此需分两种情况讨论服务质量的匹配。将第一类质量指标称为积极指标, 第二类质量指标称为消极指标。

积极指标按公式(1)计算, 消极指标按公式(2)计算得到:

$$V_{ij} = \begin{cases} \frac{Q_{i,j} - Q_j^{\min}}{Q_j^{\max} - Q_j^{\min}} & \text{如果 } Q_j^{\max} - Q_j^{\min} \neq 0 \\ 1 & \text{如果 } Q_j^{\max} - Q_j^{\min} = 0 \end{cases} \quad (1)$$

$$V_{ij} = \begin{cases} \frac{Q_j^{\max} - Q_{i,j}}{Q_j^{\max} - Q_j^{\min}} & \text{如果 } Q_j^{\max} - Q_j^{\min} \neq 0 \\ 1 & \text{如果 } Q_j^{\max} - Q_j^{\min} = 0 \end{cases} \quad (2)$$

公式中,  $Q_j^{\max}$  代表某项质量参数指标的最大值,  $Q_j^{\min}$  代表某项质量指标的最小值,  $1 \leq i \leq n, 1 \leq j \leq 3$ , 其中  $i$  表示第  $i$  个服务,  $j$  表示第  $j$  个质量指标。

为了比较请求服务质量参数和发布服务质量参数对应指标的差异值大小, 这里的  $Q_j^{\max}$ 、 $Q_j^{\min}$  不是单独指发布服务中的最大值和最小值, 而是包含请求服务的质量参数在内的最大和最小值。 $Q_{i,j}$  指的是各个发布服务中的对应服务质量指标。

得到对应的标准后, 可对每个服务进行规范化。规范化公式为:

$$\text{Score}(s_i) = \sum_{j=1}^3 (V_{ij} * W_j)$$

式中  $W_j$  是服务使用者加在各项质量指标上的一个权重, 满足:  $W_j \in [0, 1], \sum W_j = 1$ 。

举个例子来说: 服务请求者对所查找服务的 QoS 要求为 (500, 100, 0.8); 现有三个广告服务 Adv1、Adv2、Adv3, 它们的质量指标分别为 (50, 20, 0.9)、

(60,45,0.8)、(60,30,0.9)。这里假设三个质量指标的权重分别为0.3,0.3,0.4。

依据公式(1)、(2)可得:

对 Adv1:  $V_{11} = 1, V_{12} = 1, V_{13} = 1$ ;

对 Adv2:  $V_{21} = 44/45, V_{22} = 11/16, V_{23} = 0$ ;

对 Adv3:  $V_{31} = 44/45, V_{32} = 7/8, V_{33} = 1$ ;

所以  $QoSMatch(Adv1, Req) = Score(s_1) = 1 \times 0.3 + 1 \times 0.3 + 1 \times 0.4 = 1$ ;  $QoSMatch(Adv2, Req) = Score(s_2) = 44/45 \times 0.3 + 11/16 \times 0.3 + 0 \times 0.4 = 0.4996$ ;  $QoSMatch(Adv3, Req) = Score(s_3) = 44/45 \times 0.3 + 7/8 \times 0.3 + 1 \times 0.4 = 0.9558$ 。

假设用户设定的服务质量相似度阈值为0.75,则返回的满足用户需求的服务列表次序为 Adv1、Adv3。

#### 4 实验结果与分析

为了测试和验证匹配方法,实验中提供了图形化界面,通过指定对各个阶段的最小匹配等级,然后按下 <Match> 按钮开始进行服务匹配,当期望得到的匹配等级都小于服务匹配方法推出的匹配等级时,返回 true,即满足所期望的匹配结果。

通过实验分析,该匹配算法提高了服务的查准率和查全率。

与普通的基于属性值的服务查找相比,文中的服务查找增加了语义信息,使得找到的服务与实际需要的服务更加符合,从而增加了查准率。

文中首先在本体的语义表达部分引入了等价和子类关系,使得在匹配中相关本体中的实体可以匹配,增加了可能匹配服务的范围,使得服务的查全率有所提高。

#### 5 结束语

介绍了语义 Web 服务的描述语言 OWL-S,设计并实现了一个基于普适计算的服务发现系统。针对 OWL-S 描述的服务的发布和查询过程进行了介绍,重点介绍了服务匹配算法。通过定义服务发布者和请求者共同遵守的本体和服务的语义描述,达到可以实现语义层次的匹配,从而为服务与服务之间的自动组合和协作奠定了基础,使得动态地利用网络中的资源,即时组合出新的应用成为可能。

#### 参考文献:

- [1] 徐光祐,史元春,谢伟凯. 普适计算[J]. 计算机学报,2003,26(9):1042-1050.
- [2] OWL Web Ontology Language Guide[R/OL]. 2003. W3C Candidate Recommendation 18 August 2003. <http://www.w3.org/TR/2003/CR-owl-guide-20030818/>.
- [3] Martin D. OWL-S: Semantic Markup for Web Services[R/OL]. 2004-02. Technical report, Daml consortium, <http://www.daml.org/services/owl-s/1.0/owl-s.pdf>, WRSP Primer Working Draft 0.3.
- [4] 赵军,基于 OWL-S 的 Web 服务发现系统的研究和实现[J]. 计算机技术与发展,2006,16(10):163-166.
- [5] Kopena J, Regli W. DAMLJessKB: A tool for reasoning with the semantic web[J]. In IEEE Intelligent Systems, 2003, 18: 74-77.
- [6] Mokhtar S B. EASY: Efficient semantic Service discovery in pervasive computing environments with QoS and context support[J]. J. Syst. Software, 2007, 10: 9-11.
- [7] 仲梅. 基于语义的 Web 服务发现[D]. 镇江: 江苏大学, 2006: 35-37.

(上接第 129 页)

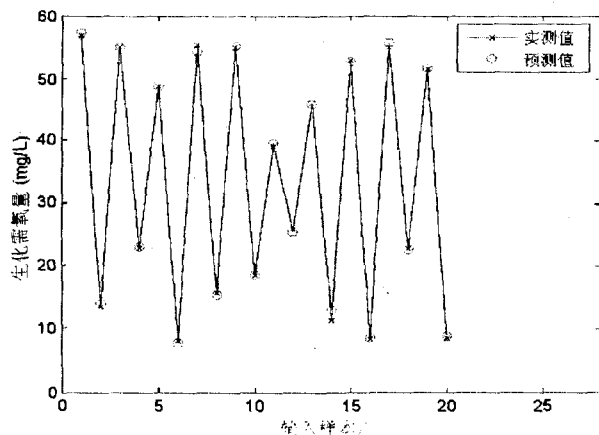


图4 BOD预测情况

效性,能实现较高的预测精度。该模型的实现将进一步推动污水处理过程实时闭环控制的实现和开发其他

工业过程的具有智能优化算法的软测量仪表具有一定的启发意义。

#### 参考文献:

- [1] 邹家庆. 工业废水处理技术[M]. 北京: 化学工业出版社, 2003.
- [2] 于静江,周春晖. 过程控制中的软测量技术[J]. 控制理论与应用, 1996, 13(2): 137-144.
- [3] 欧阳林群. 基于 GA-BP 的个股超短线预测应用研究[J]. 微计算机信息, 2007, 1(2): 306-307.
- [4] Whitley D, Starkweather T, Bogart C. Genetic algorithms and neural networks: Optimizing connections and connectivity[J]. Parallel Compute, 1990, 4(3): 347-361.
- [5] 张庆红,程国建. 基于遗传算法的神经网络性能优化[J]. 计算机技术与发展, 2007, 17(12): 125-127.