

基于领域本体的可信服务组合*

王海艳^{1,2}

(1. 东南大学 计算机科学与工程学院, 南京 211189; 2. 南京邮电大学 计算机学院, 南京 210003)

摘要: 针对现有 Web 服务组合方法缺乏对服务自身属性信任问题的深入考虑, 服务组合过程可信需求无法得到保障, 提出了一种基于领域本体的可信 Web 服务组合方法。将服务的非功能属性分为服务质量、服务信任和服务上下文三个主要方面, 并采用本体构建工具 Protégé 构建原子服务本体, 利用推理机 FaCT++ 对原始服务集进行本体推理过滤; 引入了融合本体推理过滤的 Web 服务架构; 提出了基于层次任务网络(hierarchical task network, HTN)规划法的可信服务组合算法。仿真实验表明: 与传统的组合方法相比, 该方法可有效提高服务组合的效率和成功率。

关键词: Web 服务; 领域本体; 可信; 服务组合

中图分类号: TP393.08

文献标志码: A

文章编号: 1001-3695(2012)11-4193-06

doi:10.3969/j.issn.1001-3695.2012.11.049

Trustworthy Web services composition based on domain ontology

WANG Hai-yan^{1,2}

(1. School of Computer Science & Engineering, Southeast University, Nanjing 211189, China; 2. School of Computer Science, Nanjing University of Posts & Telecommunications, Nanjing 210003, China)

Abstract: Most current Web services composition models do not take trust into great consideration. The process of service composition can not guarantee the requirements of trustworthiness. To address this problem, this paper proposed a trustworthy Web services composition method based on domain ontology. Three main components including QoS, trust of service and context composed non-functional properties of service. Protégé ontology editor built atomic service ontology and an inference engine FaCT++ helped filter original Web services. After presenting ontology inference filter-integrated service architecture, this paper introduced a trustworthy services composition algorithm based on hierarchical task network (HTN) planning. Simulation experiments show that, compared with traditional methods, the proposed method can effectively improve both the efficiency and success rate of service composition.

Key words: Web services; domain ontology; trustworthiness; service composition

0 引言

Web 服务是一种自适应、自描述、模块化、面向服务的 Internet 应用, 其高度的互操作性、跨平台性和松耦合等特点特别适合于连接跨企业边界的业务过程, 在学术界和产业界得到了广泛关注。然而, 现实中的 Web 应用一般都非常复杂, 为了分散和简化应用逻辑, 提高服务可重用性和执行的高效性, 单个 Web 服务往往功能比较简单, 复杂服务的应用需要组合多个简单的 Web 服务^[1]。

由于 Web 服务利用基于 XML 的开放标准和广泛使用的网络协议实现了面向服务的体系结构, 不仅需要解决异构分布式计算以及代码与数据重用等问题, 而且在服务交互协议方面需要提高用户的可信度。然而: a) 服务提供商所提供的服务描述信息错误和不足; b) 服务的功能、参数或者属性在动态演化过程中可能改变, 使得服务不再满足最初用户的需求, 造成服务的可用度降低; c) 服务提供商在发布注册服务信息之后, 不能主动对服务的信息进行实时更新, 使得注册前本地服务信息与注册后服务代理中的服务信息不匹配; d) 服务质量(qua-

lity of service, QoS) 属性的可信度没能在 UDDI 中得到有效验证。除此之外, 服务资源的区域性和服务参与者的社会网络性, 都会产生服务自身的信任问题^[2]。本文利用本体工具解决服务组合中的信任问题, 提高服务组合的可信性及成功率。

1 相关工作

文献[3]指出 Web 服务基本架构描述为三个参与方和三个基本操作组成, 如图 1 所示。三个参与方分别是服务请求方、服务提供方和服务注册代理中心, 三个基本操作是查找、发布和绑定。服务提供方将其服务发布到服务注册代理中心上; 当服务请求者对该服务提出请求时, 他首先利用服务代理提供的目录去搜索该服务, 得到如何调用该服务的信息; 然后根据这些信息去调用服务提供者发布的服务。服务信息的通信是在服务请求者和提供者之间直接进行, 无须经过服务代理。Web 服务体系使用一系列标准和协议实现相关的功能: 使用 WSDL(Web service description language)来描述服务, 使用 UDDI(universal description, discovery, integration)来发布、查找服务, 而 SOAP(simple object access protocol)被用来执行服务

收稿日期: 2012-04-26; **修回日期:** 2012-06-02 **基金项目:** 国家自然科学基金青年科学资助项目(61201163); 江苏省自然科学基金滚动资助项目(BK2011072); 江苏省博士后科研资助计划项目(1002005C)

作者简介: 王海艳(1974-), 女, 江苏盐城人, 教授, 博士(后), 主要研究方向为服务计算、可信计算技术(wanghy@seu.edu.cn)。

调用。

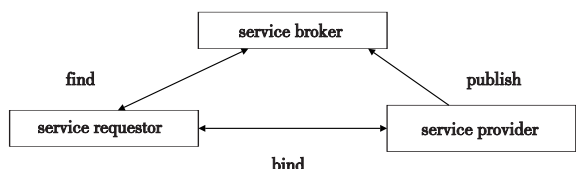


图1 经典服务架构

一般地,服务框架根据服务的功能属性进行选择,再根据服务的非功能属性进行排序,去获得满足请求者的服务序列组合。在动态开放的服务计算环境中,任何服务提供者都可以提供自己的服务,服务请求者就面临着如何在具有相似功能的庞大服务集中选取满足需要的可信服务的问题。同时,获取的服务不仅要在功能上得到满足,服务的非功能特性如服务质量、服务的上下文和服务的可信程度等也会限制服务的获取。三者相互制约,任何一方的变化都可能影响服务模型的总体性能。

文献[4]提出并设计了一种基于虚拟服务的可信保障机制,但没有具体分析服务自身可信属性,也没有涉及服务上下文等方面的研究。

文献[5]给出一个基于中间件的服务选择平台的服务组合构架,采用包含 QoS 驱动的服务选择和服务组合的 Web 服务质量模型,但缺少对服务信任的研究。

文献[6]从 QoS 数据来源的角度对服务质量属性进行分类和计算,对数据来自服务提供者的质量属性,使用以往运行数据统计,对提供者的 QoS 数据进行修正;对于数据来自服务使用者的质量属性,通过计算用户间以往反馈的相似程度权衡不同 QoS 反馈数据的可信程度,但其缺少对服务上下文属性的分析。

文献[7]根据重规划方法,监控运行时的服务 QoS 感知,提出一种 QoS 感知的重规划服务组合方法,但 QoS 的可信度没有得到保证,且没有服务上下文的分析。

文献[8]给出一种基于上下文策略匹配的 Web 服务模型,通过对服务请求方和提供方的上下文进行建模,提出一种由上下文描述和上下文策略组成的两层 Web 服务上下文模型。然而,服务自身的信任问题以及上下文数据的信任问题没有有效的保障措施,且没有考虑服务质量属性。

文献[9]给出一种基于语义的、上下文感知的动态服务组合框架,采用中间件技术和语义组件工作流的机制,实时感知用户的上下文,并根据用户上下文和偏好组合成不同的应用,其中用户的偏好的用户自身的具体规则和机器学习两种方式获得。虽然采用语义技术使得用户可以用自然语言进行请求提出,但只是笼统地分析上下文,忽略了服务上下文和资源上下文,也没有涉及服务自身及其服务质量的信任问题。

综上所述,在开放的网络环境下,现有的基于 QoS 或上下文的服务组合方法大都假设服务提供者给出的服务属性数据是真实可信或粗略修正后可信。然而,服务提供者的主观因素往往不能保证服务信息的完全真实性,甚至会导致服务质量的管理失控现象。针对现有的服务组合模型对服务信任问题考虑不够深入,本文重点研究基于领域本体实现服务的推理过滤和可信组合,在原子服务和组合服务两个服务层面上保证服务的可信。主要工作如下:

a) 构建包含可信属性的服务本体过滤组件,保证原子服

务进行可信推理,从原子服务层次上实现服务的可信保证;

b) 对本体推理过滤后的原子服务和先前状态组合后保存的过时复合服务组成的服务集进行可信组合,在组合后的复合服务层次进一步保证服务的可信。

在服务选择和组合处理上增加可信属性,使得服务请求方和满足可信要求的服务提供方在服务层面建立信任,从而在可信服务协议保障双方可信交互的前提下,在原子服务和复合服务两个层面保证服务的可信性。

本文结合领域本体知识构建包含可信属性类、上下文属性类和 QoS 属性类的 Web 服务本体,利用本体工具 Protégé 及其针对领域本体的推理机 FaCT++ 实现服务推理过滤,并探讨服务请求者信任需求,通过实例分析可信服务的选取。引入一种改进的 Web 服务架构,增加包含服务本体的服务过滤模块,采用基于层次任务网络的可信服务组合算法对推理过滤后的服务进行可信组合,从而在原子服务和复合服务两个层面保证服务请求的可信反馈,并对可信组合进行仿真实验分析。

2 包含信任属性的服务本体

现有的 Web 服务的协议和标准大多仅限于语法层次,不能很好地表达服务的语义信息。领域本体通过对特定领域的概念及概念间关系的精确描述,并通过捕获相关的领域知识,提供对该领域知识的共同理解,确定该领域内共同认可的概念,给出这些概念和概念之间相互关系的明确定义,减少了由于概念理解的歧义所产生的错误和失误。Web 服务领域的本体形成服务领域的知识表达系统核心,利用本体抽象的概念和术语,可以清楚分析 Web 服务领域的知识结构,为下文 Web 服务过滤和组合提供自动化语义支持。

2.1 服务本体相关定义

定义 1 领域本体可用五元组表示:

$$O = \{C, P, R, HC, AO\} \quad (1)$$

其中: C 为概念的集合; P 为属性的集合,本体的属性具有作用域(domain)和值域(range),体现的是概念间的二元关系; R 为关系的集合; HC 表示概念层次,即继承关系集,包括概念继承和属性继承; AO 为本体规则集,用一阶谓词逻辑或描述逻辑形式化地进行描述^[10]。

定义 2 原子服务 $WS = (S_x, R_i, P_j)$ 。其中: S_x 表示服务名; R_i 表示请求者; P_j 表示提供者,这三个参数决定一个特定的原子服务。

现有的多数服务本体都是针对某一领域特定的服务,或是仅对服务质量进行本体构建,或是只对服务上下文本体进行分类,或是仅仅分析服务的可信度。本文综合服务的这三个主要非功能属性,构建 Web 服务技术领域通用的服务本体,把服务本体表示为一个由上下文、服务质量和信任组成的三元组。

定义 3 原子服务本体是由服务上下文、服务质量和信任组成的三元组:

$$ASO(\text{atomic service ontology}) = \{\text{context}, \text{QoS}, \text{trust}\} \quad (2)$$

其中:QoS 是一个可扩展的动态向量,本文主要从价格、可用性、网络流量和响应时间四方面进行研究;context 根据服务模型和服务交互场景,可把上下文分为提供方上下文和请求方上下文,提供方上下文又分为功能和非功能上下文,非功能上下文主要分为若干个 QoS 属性上下文和信任上下文;trust 分为

服务提供方信任、服务注册中心信任和服务 QoS 信任。根据文献[11]服务请求方对服务提供方和注册中心的信任可以形式化分类和推理实现可信验证,文中不再细述。服务 QoS 信任需根据用户对 QoS 的特定需求(如价格高低、网络流量限制的要求等),在特定的非功能 QoS 上下文中利用信任值计算方法计算得到。

在 Web 服务技术的具体领域,为了获取优质的原子服务,除了满足用户的功能需求之外,还需要全面地考虑 Web 服务的非功能属性。下面对 Web 服务的上下文因素、服务质量和信任三方面进行分析建模,并进行本体推理过滤。

定义 4 在领域本体中,如果概念 C_j 被定义为概念 C_i 的 `equivalentClass`,则 C_j 和 C_i 是语义相等的,记为 $C_i \equiv C_j$ 。

假如需要测试 QoS 中 `cost` 属性是否是在一定上下文下可信的 `cost`,那么只要设置 `cost` 和用户特定需求是 `equivalentClass` 的,然后根据已有的信任计算方法计算获得 `CostTrust` 值和上下文捕捉或匹配获得 `CostContext`,就能找到满足用户特定需求的 `cost` 的原子服务本体 $\{Cost, CostContext, CostTrust\}$,即在一定上下文可信的 `cost`。

定义 5 在领域本体中,如果概念 C_j 定义为概念 C_i 的 `subClassOf`,则 C_i 语义包含 C_j ,记为 $C_i \supseteq C_j$ 。

如 Web 服务技术领域里的 `cost`、`availability`、`NetFlow` 和 `ResponseTime` 等都是 QoS 的 `subClassOf`。

语义相等和语义包含是构建本领域体的基本规则,反应概念之间的层次关系^[12]。根据这两个基本规则进行服务本体的构建。

本文采用的本体工具 `Protégé` 是由斯坦福大学基于 Java 语言开发的本体编辑和知识获取开源软件,主要用于语义网中本体的构建,是语义网中本体构建的核心开发工具。它提供了本体概念类、关系、属性和实例的构建,并屏蔽了具体的本体描述语言,用户只需在概念层次上进行领域本体模型的构建即可。它可以通过用户定制实现 Web 服务技术领域相关的支持,用于创建 Web 服务知识模型并填充用户需求的数据。针对领域本体的推理机 `FaCT++` 是以描述逻辑为理论基础,自带规则库,采用经典的 `Tableau` 算法,嵌入在本体构建工具 `Protégé` 上,使用方便效率高。

原子服务本体是一个信息过滤检索本体,通过满足服务上下文、服务质量和信任三个子类的要求,过滤所需服务信息,进而定位到满足要求的服务。例如,假设用户需求为某一响应时间内具有一定功能的服务,可不考虑其他因素,根据服务本体过滤 `ResponseTime` 类,从具有相同功能的服务集中选择上下文匹配确定的 `ResponseTimeTrust` 值最高的服务。`ResponseTimeTrust` 的计算可通过下文给出的原子服务信任计算方法计算获得,上下文 `ResponseTimeContext` 的需求可根据文献[8]上下文策略引擎获取 `Provider` 的上下文状态与之进行匹配。然后再选择一定 `ResponseTime` 值的服务,就能定位到响应时间一定的可信服务 $SW(S_x, R_i, P_j)$ 。具体匹配方法可参见文献[13],在此就不再讨论。若用户还有其他需求,可依次过滤,直到满足用户所有属性需求。

2.2 服务信任本体实例分析

参考本体构建方法,通过实例分析如何通过服务信任本体选择可信的原子服务。分析 Web 服务的信任属性,对 Web 服

务的信任属性进行本体构建,进而说明如何支持 Web 服务可信组合的实现。

在不同的上下文中信任具有不同的涵义。一般地,在开放网络环境中 Web 服务的信任可分为三类^[14],即基于策略的、基于声望的和基于权威第三方的。其中,基于策略的信任根据具体的条件创建规则集,进而判断所申请资源的可信程度,其多是考虑到 Web 服务的安全问题。而考虑声望的信任可总结表述为社交网络的信任中心化问题。至于第三方权威机构多是基于扩展的可信实体进行信任评估。本文对原子服务的信任过滤是采用基于策略的本体推理过滤。上文中的 Web 服务的信任本体(Web service trust ontology, WSTO)实质上给出了一种 Web 服务领域应用程序的信任管理方法。信任在本文表示满足一定上下文的 Web 服务属性的可信程度。为了满足用户的信任需求,本文把基于信任的 Web 服务筛选归为一个分类问题,Web 服务根据具体用户请求对应的具体策略进行分类。分类是按照一定的匹配标准去寻找满足既知的服务属性状态的用户请求策略,把满足用户请求策略的 Web 服务进行归类,并给出这些可信服务的信任认证集。其中,策略表示一系列的信任请求和信任认证。

Web 服务根据一定的匹配标准将用户请求策略和已知服务的属性状态进行匹配。如果一个策略可以满足一些上下文的服务属性状态,而与其他任何条件都不符合,称之为有效匹配。一般地,在开放网络环境中需要满足 Web 服务所涉及的所有上下文状态的用户请求策略,称为完全匹配。既知的服务属性状态,用 $\langle f, v \rangle$ 表示,其中 f 表示服务属性, v 表示该属性的值;用户请求策略是一个服务属性特征的有限集,用 $\langle f, c \rangle$ 表示,其中 c 表示服务属性的上下文。如果属性值 v 满足一定上下文 c ,则两者匹配,即该属性是可信的。

下面通过一个旅游代理服务的例子说明服务请求策略和服务属性的匹配,选取满足用户具体需求的可信服务选择。例如,用户需要获得某一天南京到上海的列车时刻表,用户可以将自己的用户文档发布到 IRS-III 上^[15],其中包括用户的信任请求和信任认证信息。用户文档和 Web 服务信任文档如文档 1、文档 2 所示。

文档 1 用户文档

```
USER1
def-class trust-profile-USER4 (trust-profile)
{
  (has-trust-guarantee :type guarantee-USER1)
  (has-trust-requirement :type requirement-USER1)
}
def-class requirement-USER1 (security-requirement)
{
  (encryption-algorithm :value high)
  (certification-authority :value high)
}
```

文档 2 Web 服务信任文档

```
TRUSTED WEB SERVICE WS1
def-class get-train-timetable-service-T1 (trust-Web-service)
{
  (has-capability :value get-traintimetable-capability-T1)
  (has-interface :value get-traintimetable-service-interface-T1)
  (has-trust-profile :type get-traintimetable-service-trust-profile-T1)
}
def-class get-train-timetable-servicetrust-profile-T1 (trust-profile)
{
  (has-trust-guarantee :type get-traintimetable-service-guarantee-T1)
  (has-trust-requirement :type get-traintimetable-service-requirement-
```

```

T1)
{
    def-class get-train-timetable-serviceguarantee-T1 ( Trust-non-functional
properties)
{
    ( encryption-algorithm ;type RSA)
    ( certification-authority ;value verisign))
}

```

假设该实例包括的用户和 Web 服务所对应的策略具有相同的参数,都包括加密算法和 CA(certification authority)。预先规定 DES、AES、MD5、RSA 等加密算法或 CA 认证模式的等级,如加密算法 RSA 和电子签名的等级最高。而在 Web 服务的文档信任认证集中 Web 服务 WS1 的加密算法是 RSA,认证模式是电子签名,则可知 Web 服务 WS1 可信,从而获得用户可信的列车时刻表查询服务。

3 融合本体推理过滤的 Web 服务架构

由于经典服务架构缺少必要的服务语义信息以及对服务属性数据信任的处理,基于上文已经增加的服务本体过滤组件,对原始服务集进行本体过滤,提出一种改进的 Web 服务架构。改进模型除了包含传统 Web 服务的三类基本模块,即服务请求模块、服务注册模块和服务提供模块,另外还增加了原子服务本体过滤组件模块,具体构成如图 2 所示。

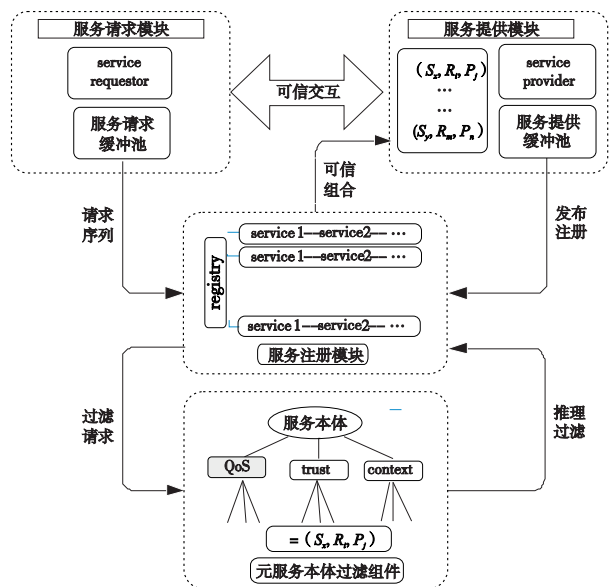


图2 融合本体推理过滤的Web服务架构

服务注册模块是接收服务提供模块注册发布服务和响应服务请求模块请求的第三方模块。服务本体过滤组件根据服务请求模块的请求序列,根据服务非功能属性建立的服务本体对发布注册的原子服务进行推理过滤,使之满足既定需求。其中,原子服务本体包含上下文属性子类、信任属性子类和 service 质量属性子类。上下文属性或根据需要可选,或服务注册中心实时捕捉;信任属性根据已有的信任计算方法可以获得;服务质量属性根据请求方特定需求而定。

服务请求方和服务提供方增加服务缓冲池,缓冲池对服务进行排队,让服务形成序列有次序的进出,提高服务调度效率;服务提供模块完成可信组合任务,接收原子服务本体过滤组件推理出的满足特定需求的原子服务到服务集中;服务集还包括先前状态保存下来的复合服务,如图 3 所示。综上所述,在服务自身层面形成可信组合服务链,进而与服务请求模块在服务

层面建立可信交互连接。

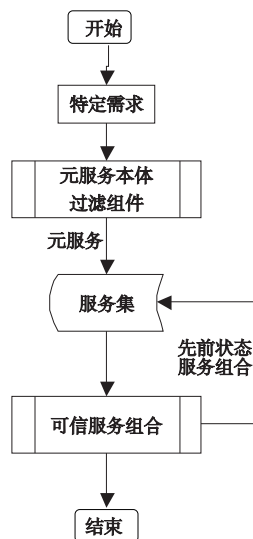


图3 改进的服务组合流程

服务提供方向服务注册模块发布注册服务,将发布的服务挂靠在对应的 provider 后,服务注册中心反馈发布信息。服务请求方提出服务请求,通过服务请求缓冲池调用请求序列到服务注册中心。服务注册中心根据请求方的个性化需求向服务本体过滤组件提出过滤请求,过滤组件经过服务本体进行推理,过滤反馈结果给服务注册中心。服务注册中心根据过滤反馈后的原子服务向服务提供方提出可信组合请求,并要求建立可信交互。至此,服务请求方和服务提供方在服务层面建立可信交互。具体的服务消息机制如图 4 所示。

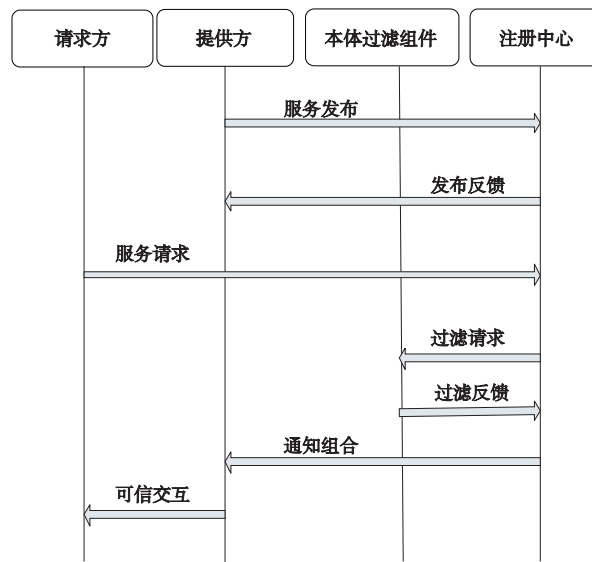


图4 改进的服务消息机制

4 可信服务组合方法

上文中改进的 Web 服务架构利用领域本体技术,对 Web 服务主要的非功能属性进行形式化分析和必要的语义信息描述,对原始服务集进行自动化过滤处理。然而,单个原子服务提供的功能有限,需要将原子服务进行组合。下文介绍通过可信原子服务组合服务的形式,在提供方形成可信服务链响应服务请求者。

定义 6 复合服务。具有单个功能的原子服务组合成的链,形成满足复杂功能的服务集合。

定义 7 服务集 W 是由服务提供方在服务注册中心发布后还没处理的原子服务或已经组合遗留下来的复合服务的集合。

4.1 服务信任值计算

假定 w 是原子服务; F 为服务的属性集; ρ 为用户等级, 即用户对某一服务属性的评价等级。如用户对服务 w 价格 cost 的评价可表示为 $\rho(w, \text{cost})$ 。设 R 为用户信任等级的有限集合, U_ρ 是服务 w 的所有用户。对于 $\forall u \in U_\rho$, 都有 $\rho_u \in R$ 。 $F_c(w)$ 是请求者 c 对服务 w 的偏好属性集合。请求者 c 对服务 w 的属性 f 的信任度可表示为

$$t_c(w, f) = \sum_{u \in U_\rho} \rho_u(w, f) \times \text{trust}(c, u) \quad (3)$$

其中: $\text{trust}(c, u)$ 是请求者对提供者的信任度, 可根据实体信任计算方法获得。那么, 请求者 c 对服务 w 的信任度就可以表示为

$$t_c(w) = \frac{\sum_{f \in F_c(w)} t_c(w, f)}{|F_c(w)|} \quad (4)$$

如果 w 是复合服务, 根据层次任务网络规划法的思想^[16], 将服务分解成有若干个子处理组成的处理过程。设实现复合服务 w 的处理过程的集合 M 被分为 m 个子处理过程, 则复合服务的信任值可表示为

$$t_c(w) = \max_m t_c(m, w) \quad (5)$$

其中: $t_c(m, w)$ 指子处理过程 m 中的信任度, 选取所有处理过程中的信任值最大值作为复合服务 w 的信任值。最后, 把计算的服务信任值都存放在服务信任值表 θ 中。

4.2 可信服务组合算法

本文通过服务本体筛选在语义上保证服务属性的可读性, 在规避机器异构问题的基础上, 采用层次任务网络(hierarchical task network, HTN)规划法的思想, 将可信服务组合转为任务问题, 找到解决这个任务问题的最优方法就可得到所需的服务组合^[17]。仅对满足服务请求的服务进行处理, 不必查找整个服务集, 从而可高效地实现基于可信服务属性的服务组合。

层次任务网络规划法利用 OWL-S 作为 Web 服务的描述语言, 通过任务分解的方法来组合服务。一个基于 HTN 的规划系统将目标任务分解为子任务的集合, 这些子任务再分解为更小的子任务, 一直循环这个过程, 直到所有的子任务仅包含原子 Web 服务。在每一轮任务的分解过程中, 都会检验是否违反给定的条件(如子任务数量是否超过一定数目)。如果目标任务在不违反任何给定的条件下, 能够被成功分解成原子任务集合, 则规划问题成功求解。

$$\text{problem} = (s, c, G, W, F, R) \quad (7)$$

其中: s 指初始状态; c 是用户; G 是目标集; W 是服务集; F 是服务属性特征集; R 是用户 ratings 有限集。解决这个问题, 找到达到目标集 G 的最大可信程度的服务组合即可完成可信服务组合。

每个服务的信任值都放在信任值表 θ 中, 初始状态的信任值表中统一对每个服务进行初始化。经过原子服务过滤模块筛选的原子服务加入到服务集后, 对那些在服务集中满足用户请求目标的服务进行服务信任值计算, 如果信任值大于表中的数值, 则把该服务加入序列 π 中并更新信任值表 θ ; 对于服务集中在之前组合任务保留下来的复合服务, 算法通过对服务在新的上下文中的属性值信任计算, 方法同原子服务。依次更新

服务组合序列和服务信任值表, 直至找到一个能够满足用户所有请求的服务序列为止。最后, 计算组合后的服务信任度并更新信任值表, 供下次来的组合需求调用。HTNTrustComposition 算法描述如下:

基于 HTN 的可信服务组合算法:

Input: 可信服务组合问题 $\text{Problem} = (s, c, G, W, F, R)$ 和空服务组合 π_0 。

Output: 可信服务组合 π 。

HTNTrustComposition(s, c, G, W, F, R, π)

begin

if $G = \emptyset$;

then return π_0 ; // 如果目标集为空, 则返回空 π_0

$g \rightarrow G$; // * 更新目标集 G , 如果请求方对服务属性的需求 g 不在 G 中, 则将它作为子目标加入到 G 中, 根据 QoS 向量的可扩展性 */

if $\exists w \in W$ can complete g under state s ;

// 在状态 s 下, g 可被原子服务 w 实现

$\forall f \in F$; // 请求方需求的 Web 服务属性在服务属性集中

$\tau(w) \leftarrow \text{ComputingTrust}(w)$;

// 计算原子服务信任值

if $\tau(w) > \theta(w)$;

then insert w into π ; // 将服务 w 插入到 π 中

Updating Web service trust table θ ; // 更新服务信任值表 θ

$\pi \leftarrow \text{TrustComposer}(s', c, G, F, \mathcal{R}, \pi)$; // 递归运算

else

$Q(w) = \text{ComputingTrust}(w)$; // 计算复合服务信任值

if $Q(w) > \theta(w)$;

then updating Web service trust table θ ;

Insert w into π ;

$\pi \leftarrow \text{TrustComposer}(s', c, G, F, \mathcal{R}, \pi)$; // 递归运算

return π ;

end

可信服务组合算法把整个服务组合过程作为一个大任务, 把需要组合的每个原子服务作为组合拆分成的子任务进行分治处理, 在保证服务可信度的前提下, 仅对满足用户需求目标集的服务进行处理, 大大提高了组合效率。

4.3 实验分析

实验在 Windows XP 系统下, 采用 MATLAB 工具进行仿真。由于目前 Web 服务还没有标准的测试平台和数据集, 本文采用模拟数据集进行测试。

实验 1 服务组合效率分析

为了说明基于 HTNTrustComposition 算法的组合方法 (HTN-based trust service composition method, HTNTSCM) 效率, 将之与基于关键字匹配 (keywords-based service composition method, KSCM)^[18] 和基于最佳路径算法的服务组合 (optimal route-based service composition method, ORSCM)^[19] 方法进行比较。基于关键字匹配服务请求的服务组合方法没有考虑服务属性的语义, 虽然组合效率比较高, 但组合质量局限性比较大。基于最佳路径算法的服务组合需要对服务组合查找树中的全部服务组合进行遍历, 当服务集服务数量很大时, 效率不明显。实验测试数据为服务集中的服务个数, 设为 0 ~ 1 000, 共分 10 组数据。

三种方法的服务集中服务的数量和服务组合效率之间关系的仿真结果如图 5 所示, 基于关键字匹配的组合方法虽然组

合效率是最高的,但没考虑语义局限组合质量。基于最佳路径算法的服务组合需要查找服务集内所有服务,随着服务集的增大组合消耗的时间急剧增长,效率很不理想。本文提出的 HTNTSCM 组合方法是基于层次任务网络的,且组合对象是针对本体语义过滤后的服务,在语义层次保证了服务属性的质量,组合效率略低于 KSCM,且随着服务集的增大组合时间缓慢增长,但远比 ORSCM 效率好,从而说明了基于层次任务网络的可信服务组合算法在保证服务可信度的情况下,具有很好的服务组合效率。

实验2 服务组合成功率分析

已有的一些服务组合方法大多没考虑服务属性的可信度,本实验通过比较考虑 Web 服务信任属性的 HTNTSCM 组合算法和没有考虑服务信任属性的基于领域本体组合方法(domain ontology-based service composition method, DOSCM)^[10]的服务组合成功率,进而验证信任关系对服务组合成功率的影响。

图6表明,随着时间的变化,考虑服务信任属性的 HTNTSCM 服务组合方法使得服务集中的服务逐渐与服务请求者建立信任关系,并记录在信任值表 θ 中,成功率逐渐提高,并趋向一个稳定的水平。而没有考虑信任属性的 DOSCM 方法服务组合成功率明显具有波动,呈现不稳定性。

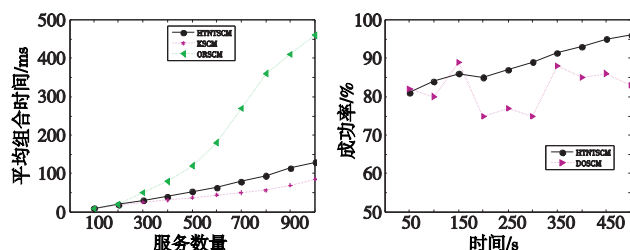


图5 服务组合效率的比较

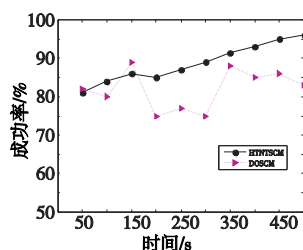


图6 组合成功率的比较

如果存在恶意节点,那么不同的组合方法会产生不同的组合成功率。恶意节点就是指那些发布的服务属性值与真实值存在巨大偏差的服务,如上下文信息或 QoS 信息与真实值相差较大。图7是恶意节点对组合成功率的仿真结果对比。在恶意节点不多的情况下,考虑信任属性的 HTNTSCM 和没考虑服务信任的一般组合方法的服务组合成功率大致相同;然而,随着恶意节点数目的增加,考虑到服务节点都会在信任表 θ 中进行信任度过滤,且信任理论具有监控和评价机制,可形成近乎可信的服务集供组合任务调用,成功率下降较慢,并趋向于一个平稳水平,而 DOSCM 组合方法随恶意节点的增加,组合成功率明显下降。

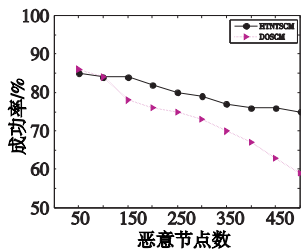


图7 恶意节点对组合成功率影响的比较

5 结束语

本文根据经典服务模型,针对现有 Web 服务组合模型中忽略对服务自身属性信任问题的研究,通过增加包含服务上下文、服务质量和信任属性的原子服务本体过滤组件,提出了一种可信的 Web 服务组合方法,即首先参考信息过滤的方

法,对原子服务进行推理过滤,进而利用基于层次任务网络规划法的可信服务组合算法,实现在原子服务和复合服务两个层面的可信保证,完成 Web 服务的可信组合和自动化处理。后期的研究工作将进一步考虑更细粒度的服务属性挖掘和服务的可信交互协议的实现。

参考文献:

- [1] 龙军,刘昕民,袁鑫攀,等.一种基于信任推理与演化的 Web 服务组合策略[J]. 计算机学报,2012,35(2):298-314.
- [2] ALRIFAI M, RISSE T. Combining global optimization with local selection for efficient QoS-aware service composition[C]//Proc of the 18th International Conference on World Wide Web. New York: ACM Press,2009: 881-890.
- [3] DUSTDAR S, SCHREINER W. A survey on Web services composition[J]. International Journal of Web and Grid Services,2005,1(1):1-30.
- [4] 詹涛,周兴社,杨刚,等.一种可信的服务组合 QoS 保障机制[J]. 小型微型计算机系统,2011,32(3):385-389.
- [5] ZENG Liang-zhao, BENATALLAH B, NGU A H H, et al. QoS-aware middleware for Web services composition[J]. Software Engineering,2004,30(5):311-327.
- [6] 李研,周明辉,李瑞超,等.一种考虑 QoS 数据可信性的服务选择方法[J]. 软件学报,2008,19(10):2620-2627.
- [7] CANFORA G, PENTA M D, ESPOSITO R, et al. QoS-aware replanning of composite Web services[C]//Proc of International Conference on Web Services. New York: ACM Press,2005:121-129.
- [8] MEDJAHED B, ATIF Y. Context-based matching for Web service composition[J]. Distributed and Parallel Databases,2007,21(1):5-37.
- [9] FUJII K, SUDA T. Semantics-based context-aware dynamic service composition[J]. ACM Trans on Autonomous and Adaptive Systems,2009,4(2):1-31.
- [10] 李曼,王大治,杜小勇,等.基于领域本体的 Web 服务动态组合[J]. 计算机学报,2005,28(4):645-646.
- [11] 王海艳,谢武峰.信任本体的形式化语义研究[J]. 武汉大学学报:理学版,2011,57(5):369-374.
- [12] Property-semantic Web standards[EB/OL]. [2011-07-12]. <http://www.w3.org/2001/sw/wiki/Property>.
- [13] HAN Li-quan, XU Zhong-yu, YAO Qing-an. An approach to Web service composition based on service-ontology[C]//Proc of the 6th International Conference on Fuzzy Systems and Knowledge Discovery. Piscataway: IEEE Press,2009:173-177.
- [14] GALIZIA P, GUGLIOTTA A, DOMINGUE P. A trust based methodology for Web service selection[C]//Proc of International Conference on Semantic Computing. Piscataway: IEEE Press,2007:193-200.
- [15] CABRAL L, DOMINGUE J, GALIZIA S, et al. IRS-III: a broker for semantic Web services based applications[J]. Journal of Web Semantics,2008,6(2):109-132.
- [16] SIRIN E, PARSIA B, WU Dan, et al. HTN planning for Web service composition using SHOP2[J]. Journal of Web Semantics,2004,1(4):377-396.
- [17] KUTER U, SIRIN E, NAU D, et al. Information gathering during planning for Web service composition[J]. Journal of Web Semantics,2005,3(2):183-205.
- [18] 艾未华,黄敬平,周宁,等.一种基于语义本体的 Web 服务自动组合法[J]. 系统仿真学报,2008,20(4):935-937.
- [19] 焦燕廷,杜玉越.一种基于领域本体的最优 Web 服务组合法[J]. 山东科技大学学报,2010,29(5):83-90.