

基于本体的 Web 服务合成研究*

韩亚洪 刘永革

(安阳师范学院计算机与信息工程学院 安阳 455000)

【摘要】 在对 Web 服务和 Web 服务合成的概念,以及 OWL-S 提供的 Web 服务上层本体和基于本体的 Web 服务合成描述进行介绍后,指出 OWL-S 对 Web 服务合成的描述能力有限,应针对 Web 服务合成进行基于本体的语义扩充。通过实例介绍如何在 WSDL 定义的概念基础上扩展得到基于本体的 Web 服务描述,在给出的 Web 服务定义的基础上讨论 Web 服务可合成性的定义,它是未来 Web 服务自动合成的前提和基础。

【关键词】 语义 Web Web 服务 服务合成 本体 **【分类号】** TP31

A Study on Web Services Composition Based on Ontology

Han Yahong Liu Yongge

(Computer and Information Engineering College, Anyang Normal University, Anyang 455000, China)

【Abstract】 This paper firstly introduces the concept of Web Services and Web Services composition. Based on the introduction of OWL-S's upper Ontology and its description for Web Services composition, the paper indicates that the capability of OWL-S's supporting for Web Service composition is limited, and the semantic expansion for composition based on Ontology is needed. By instance it introduces how to get the semantic expansion for WSDL, and based on the definition of Web Services, it discusses the definition of composability of Web Services, which is the basic for future automatic Web Services composition.

【Keywords】 Semantic Web Web Services Service composition Ontology

1 引言

参考 W3C 对 Web 服务(Web Services)的定义^[1]: Web 服务是由 URI(统一资源标识符)标识的软件系统,它的公共接口和绑定方法由 XML 定义和描述,通过定义以及基于 XML 和 Internet 协议的消息传递这些软件系统在 Web 上可以相互发现和相互调用。简单来讲,Web 服务是独立的、模块化的应用程序,能够在 Web 上被描述、发布、查找和调用。

在早期的分布式计算系统中,组件之间的紧密耦合性要求系统遵循太多的来自于不同组织的协议和标准,未来的趋势是从紧耦合的单一系统发展为松耦合的动态绑定组件系统,Web 服务正是在这种趋势下出现的。一系列基于 XML 标准的新技术,如 WSDL、SOAP 和 UDDI 的出现使得 Web 服务迅速发展并已成为目前业界和学术界研究的重点。

收稿日期:2007-04-10

收修改稿日期:2007-04-15

* 本文系河南省科技厅科技攻关项目“甲骨文数字化技术研究”(项目编号:0624480025)的研究成果之一。

一些最新的 Web 服务应用领域有:电子商务(B2B)、股票交易、电子政务等,全球最大的电子商务网站 Amazon 在 Web 服务领域(AWS:WWW.amazon.com/webservices)已经走在了前列,Amazon Web 服务(AWS)为软件开发人员提供了访问 Amazon 技术平台和产品数据的应用程序接口(API),使得开发者可以通过创建动态和高效的 Web 应用程序来更新或拓展它们的商务。

总的来说,Web 服务有两种:简单(Simple)Web 服务和合成(Composite)Web 服务。简单 Web 服务可以不依赖于其他 Web 服务独立地完成客户的要求,如检索信息的 Web 服务。合成 Web 服务被定义为简单 Web 服务或其他合成 Web 服务的聚合,它们协同工作来提供增值的(Value-added)服务内容。例如,汽车销售代理服务是一个合成 Web 服务,它由汽车经销、金融业务(如贷款)和汽车保险服务等 Web 服务合成而成。

目前,基于 XML 的 Web 服务的相关技术只为 Web 服务的描述、信息传递以及 Web 服务发现提供了语法(Syntax)上的描述,并没有明确其语义(Semantic),这就使得现在的 Web 服务合成过程非常费时,并容易出错。下一代 Web 是语义 Web(Semantic Web),其目标是为了

让计算机能够明确地解释执行任务,而这正好能够解决 Web 服务研究中的难题。语义 Web 技术的核心是基于本体的语义描述,所以基于本体的 Web 服务合成技术将能给 Web 服务合成提供帮助,并最终实现 Web 服务自动合成以及未来的智能 Web 服务。

2 基于 OWL-S 的 Web 服务合成描述

2.1 OWL-S 定义的 Web 服务上层本体

为了使用一个 Web 服务,需要一个计算机可理解的服务描述。本体 Web 语言 OWL 的目标之一就是创建这些描述被制定和共享的框架。Web 站点应该使用一个基本的类和属性的集合来声明和描述服务,这一点可以在 OWL 的 Ontology 架构下完成。OWL-S^[2] 即 Web 服务的语义标记语言,它基于本体 Web 语言 OWL,是由多个组织的研究人员联合在这方面给出的一个 Ontology,其上层本体包括 3 个部分:Service Profile、Service Process 和 Service Grounding^[2]。

Service Profile:指明所描述的 Web 服务的功能与接口,以便于服务代理能够搜索与匹配该 Web 服务;Service Process:指明当服务被调用时的操作,以便服务代理进一步匹配,以及服务合成和服务的协调工作及监控;Service Grounding:指定调用服务的具体细节,比如通信协议、调用端口等。其层次关系如图 1 所示。

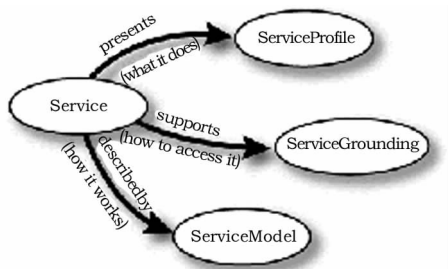


图 1 OWL-S 上层本体^[2]

例 1 给出了一个 Web 服务“ISBNFinder”的上层本体描述,该服务提供的 Web 服务是根据输入的书名找到该书的 ISBN 号。根据例 1 的描述,该服务的 Service Profile、Service Process 和 Service Grounding 分别为:ISBNFinderProfile、ISBNFinderProcessModel 和 ISBNFinderGrounding。

例 1 OWL-S 上层本体描述

```
<rdf:RDF
..... <! -- 名称空间做了省略 -- >
xmlns:service = "http://www.daml.org/services/owl-s/1.0/Service.owl#"
xmlns:profile = "http://www.daml.org/services/owl-s/1.0/Profile.owl#"
xmlns:process = "http://www.daml.org/services/owl-s/1.0/
```

```
Process.owl#"
xmlns:grounding = "http://www.daml.org/services/owl-s/1.0/
Grounding.owl#" >
<! -- 服务的上层本体描述 -- >
<service:Service rdf:ID = "ISBNFinderService" >
  <service:presents rdf:resource = "#ISBNFinderProfile"/ >
  <service:describedBy rdf:resource = "#ISBNFinderProcessModel"/ >
  <service:supports rdf:resource = "#ISBNFinderGrounding"/ >
</service:Service >
..... <! -- 省略其他部分 -- >
</rdf:RDF >
```

其中 Profile 将给出该服务的 3 种信息描述:

- (1) 服务的提供者:包含了指向提供服务的实体的联系信息;
- (2) 服务的功能:包含了 4 类信息 IOPE: Input、Output、Precondition 和 Effect,即服务的输入、输出、前置条件以及预期结果;
- (3) 服务的特征属性:服务所属的种类、服务质量评价以及参数列表。

Profile 提供了服务在注册处所需信息的准确描述,而一旦用户找到了所需的服务,后面和服务的交互通过 Process Model 进行,OWL1.0 定义了 ServiceModel 的一个子类——ProcessModel,它给出了服务的过程本体,该本体的上层描述如例 2 所示,即有 3 种类型的 Process: Atomic (原子的)、Simple(简单的)和 Composite(合成的)。

例 2 ProcessModel 的上层本体描述

```
<owl:Class rdf:ID = "Process" >
<owl:disjointUnionOf rdf:parseType = "Collection" >
  <owl:Class rdf:about = "#AtomicProcess" / >
  <owl:Class rdf:about = "#SimpleProcess" / >
  <owl:Class rdf:about = "#CompositeProcess" / >
</owl:disjointUnionOf >
</owl:Class >
```

在 OWL-S 中,ServiceProfile 和 ServiceModel 都被认为是抽象表达,只有 ServiceGrounding 是具体的实现层次的规范。OWL-S 选择了已有的工业标准 WSDL 和 SOAP 来描述这个实现规范。因为 OWL-S 也是基于 XML 的语言,所以很容易对 WSDL 做出扩展来支持 OWL-S。

2.2 OWL-S 对 Web 服务合成的描述

从 2.1 节可以看到,Web 服务上层本体的 3 个部分分别对 Web 服务提供了不同的支持,其中 Service Process 提供了对 Web 服务合成的描述。

AtomicProcess 可以直接被调用,它没有子过程,一步执行。从服务请求者看来,就是传递一个输入消息,服务执行完毕后返回一个输出消息。SimpleProcess 是不可调

用的,也没有对应的 Grounding,但是它们和 AtomicProcess 一样,被认为只有一个执行步骤。SimpleProcess 一般作为元素的抽象,它们或者提供了一些 AtomicProcess 的视图,或者是对 CompositeProcess 的简化的表达。CompositeProcess 能够被分解为其他组合的或者非组合的 Process。分解可以通过 SEQUENCE、SPLIT 或者 IF-THEN-ELSE 这样的控制结构来进行^[2]。

例 3 是一个有关“航班订票”的 Web 服务的描述,该服务 (BookFlight_Process) 由 4 个服务按照序列 (SEQUENCE,即先后关系)的控制结构来合成进行。

例 3 OWL-S 的服务合成描述实例

```
<rdf:RDF xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:process = "http://www.daml.org/services/owl-s/1.0/Process.owl#" >
  <process:CompositeProcess rdf:ID = "BookFlight_Process" >
    <process:composedOf >
      <process:Sequence >
        <process:components rdf:parseType = "Collection" >
          <process:AtomicProcess rdf:about = "#GetFlightDetails"/ >
          <process:AtomicProcess rdf:about = "#GetContactDetails"/ >
          <process:AtomicProcess rdf:about = "#ReserveFlight"/ >
          <process:AtomicProcess rdf:about = "#ConfirmReservation"/ >
        </process:components >
      </process:Sequence >
    </process:composedOf >
  </process:CompositeProcess >
</rdf:RDF >
```

这 3 个服务分别是:

- (1) GetFlightDetails,根据目的地、时间等输入信息取得航班的详细信息,这是一个信息查询的原子服务;
- (2) GetContactDetails,根据上一个原子服务得到航班订票联系方式的原子服务;
- (3) ReserveFlight,这是一个预定订票的原子服务;
- (4) ConfirmReservation,这是对预定票进行确认的服务。

由以上 4 个过程完成整个航班订票的 Web 服务。该 Web 服务合成过程可以用图 2 来表示。当然,合成服务还可以用来合成其他的合成 Web 服务。

2.3 OWL-S 存在的问题

OWL-S 基于本体定义了描述 Web 服务的语义标记,这些语义标记提供了对 Web 服务自动合成的描述,但是 OWL-S 中并未给出合成服务如何通过 OWL-S 的规格说明 (Specification) 来产生,更重要的是,OWL-S 中并未对 Web 服务的可合成性概念进行定义,而概念的定义

恰恰是 Web 服务 (自动) 合成的关键和基础。

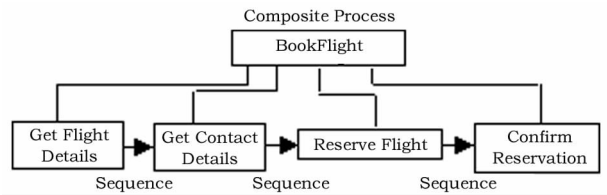


图 2 OWL-S 描述的服务合成过程

3 基于本体的 Web 服务描述和定义

Web 服务的合成需要对每一个服务进行描述,使得服务之间可以相互理解和交互。Web 服务描述语言 WSDL 是被 W3C 标准化的 Web 服务描述语言,但是 WSDL 几乎没有提供对 Web 服务的语义描述,而只是从语法角度提供了对 Web 服务特征的描述。因此,应该对 WSDL 中定义的概念进行语义扩展,即在 WSDL 定义的概念基础上扩展得到基于本体的 (Ontology-based) Web 服务描述^[3]。

这里将通过“房屋销售 Web 服务”实例给出基于本体的 Web 服务描述的定义和形式化说明。如图 3 所示,该 Web 服务由多个 Web 服务合成,它们分别是:房屋销售代理 (HB)、房屋经销商 (HD)、房屋问题检测 (HC)、银行信用卡历史 (CH)、金融支持 (FI)、保险支持 (IN) 以及房屋购买历史 (HH) 等 7 个 Web 服务。Web 服务通过服务操作调用 (Operation Invocation) 向用户或者其他服务提供本服务的功能。

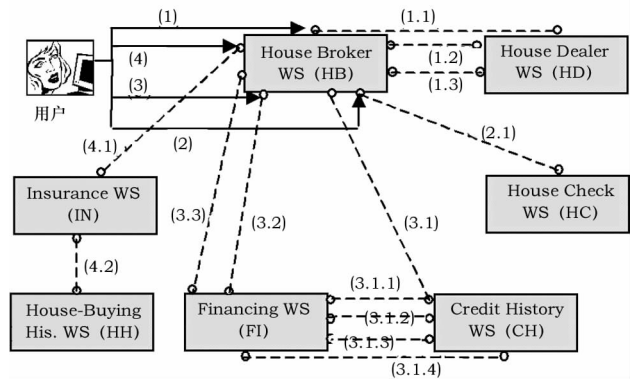


图 3 Web 服务实例“房屋销售 Web 服务”

如图 3 所示,箭头 (1) - (4) 代表用户的调用步骤,而有端点的虚线代表服务间的相互操作,每个端点处代表该端服务提供的一个操作。例如:用户的第一个调用是调用 (1),该箭头指向的端点调用了服务 HB 的一个操作 sendMePriceQuote,即需要房屋报价;该操作和服务 HD 的操作 priceQuote (虚线段的另外一个端点) 进行交互从而返回用户房屋报价。

Web 服务的操作模式 (Operation Mode) 一般有 4 种:

- (1) 请求操作 (Solicit - response OP);
- (2) 响应操作 (Request - response OP);
- (3) 单向操作 (One - way OP);
- (4) 通知操作 (Notification OP)。

例如:上文提到的操作 sendMePriceQuote 是一个请求操作;而操作 priceQuote 是一个响应操作;图 3 中服务 FI 和 CH 之间的相互操作(3.1.4)左右端分别为 notifyCustomerCredit(通知客户信用卡)是一个通知操作;receiveCustomerCredit(收到客户信用卡)是一个单向操作。

每一个操作的输入 (Input)/输出 (Output) 消息 (Message) 依操作模式而不同,请求操作和响应操作输入输出消息都有,通知操作只有输出消息,而单向操作只有输入消息。每一个消息又包含多个参数(即 WSDL 中的 parts),每个参数有名字 (Name) 和数据类型 (Data Type, 描述时一般采用 XML Schema 中内置的数据类型)。

基于以上讨论,Web 服务的操作可以定义如下:

定义 1:Web 服务的操作 (Operation)。一个操作 op_{ik} 可以定义为一个六元组 ($Description_{ik}$ 、 $Mode_{ik}$ 、 In_{ik} 、 Out_{ik} 、 $Purpose_{ik}$ 、 $Category_{ik}$),其中:

- ① $Description_{ik}$ 是操作 op_{ik} 特征的一个文本描述;
- ② $Mode_{ik} \in \{ "one - way", "notification", "solicit - response", "request - response" \}$;
- ③ In_{ik} 、 Out_{ik} 是输入输出消息;当 $Mode_{ik} = \{ "notification" \}$ 时 $In_{ik} = \Phi$;当 $Mode_{ik} = \{ "one - way" \}$ 时 $Out_{ik} = \Phi$;
- ④ $Purpose_{ik}$ 是 op_{ik} 的操作目的,被定义为一个三元组 (Function, Synonyms, Specialization),其中,Function 是操作的商业功能,由分类信息的本体 (Ontology) 定义, Synonyms 是可选的功能名称, Specialization 是功能特征说明;
- ⑤ $Category_{ik}$ 是 op_{ik} 的操作分类,被定义为一个三元组 (Domain, Synonyms, Specialization),其中,Function 是操作提交的目标服务领域,由分类信息的本体 (Ontology) 定义, Synonyms 是可选的领域名称, Specialization 是领域特征说明。

基于以上定义,给出操作 HB5 :: sendMePriceQuote 的定义实例:

例 4 操作 HB :: sendMePriceQuote 被定义为一个六元组 (Desc, Mode, In, Out, P, C),其中:

- ① Desc = "该操作返回所查房屋的参考价格"; Mode = "solicit - response";
- ② Inik = { "价格" }; Outik = { "房屋编号", "价格" };
- ③ 是该操作的操作目的定义: P. Function = { "估价申请" }, P. Specialization = { "向用户提供信息" }, P. Synonyms = { "房屋报价" };
- ④ 是该操作的操作分类定义: C. Domain = { "房屋经销商" }, C. Specialization = { "提供现房和期房报价" }, C. Synonyms = { "房产销售商" }。

如图 4 所示, Binding 和 Operation 是 WSDL 中提供的 Web

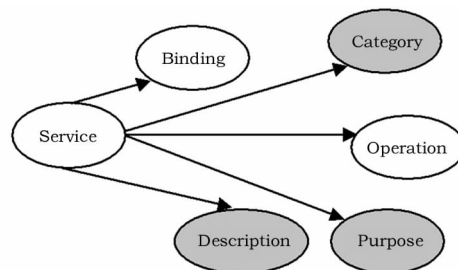


图 4 基于本体的 Web 服务描述

服务描述原语(在这里作为本体概念,增加了语义上的扩充), Category、Description 和 Purpose 是增加的对 Web 服务描述的本体概念。因此,一个 Web 服务可以定义如下:

定义 2:Web 服务。一个 Web 服务 WS_i 可以定义为一个五元组 ($Description_i$ 、 OP_i 、 $Bindings_i$ 、 $Purpose_i$ 、 $Category_i$),其中:

- ① $Description_i$ 是 Web 服务 WS_i 特征的一个文本描述;
- ② OP_i 是 WS_i 所提供的操作集合;
- ③ $Bindings_i$ 是支持 WS_i 的绑定协议集合;
- ④ $Purpose_i$ 是 WS_i 的操作目的集合;
- ⑤ $Category_i$ 是 WS_i 的分类信息集合。

基于以上定义,给出图 3 中 Web 服务 HD 的定义实例:

例 5 操作 HD 被定义为一个五元组 (Desc、OP、B、P、C),

其中:

- ① OP 是该服务所能提供的操作定义,包括 3 个操作: $OP = \{ priceQuote, testQuote, specialOffers \}$;
- ② B 是支持该服务的绑定协议集合,定义为 $B = \{ "SOAP" \}$,即采用 SOAP(简单对象访问协议)协议;
- ③ B 和 P 分别是该操作的操作目的和操作分类定义,定义方法同例 1。

4 Web 服务可合成性定义

基于以上讨论,Web 服务的可合成性通过以下 3 个定义完成:

定义 3:操作模式可合成。两个操作 op_{ik} 和 op_{jl} 是模式可合成的仅当:

- ① $Mode_{ik} = \{ "notification" \}$ 并且 $Mode_{pl} = \{ "one - way" \}$;
- ② $Mode_{ik} = \{ "one - way" \}$ 并且 $Mode_{pl} = \{ "notification" \}$;
- ③ $Mode_{ik} = \{ "solicit - response" \}$ 并且 $Mode_{pl} = \{ "request - response" \}$;
- ④ $Mode_{ik} = \{ "request - response" \}$ 并且 $Mode_{pl} = \{ "solicit - response" \}$ 。

定义 4:绑定协议可合成。两个 Web 服务 WS_i 和 WS_j 是绑定协议可合成的仅当: $Bindings_i \cap Bindings_j \neq \Phi$ 。

定义 5:操作语义可合成。两个操作 op_{ik} 和 op_{jl} 是语义可合成的仅当满足以下两个条件:

- ① ($Category_{ik} \cdot Domain = Category_{jl} \cdot Domain$) 或者 ($Category_{ik} \cdot Domain \in Category_{jl} \cdot Synonyms$) 或者 ($Category_{jl} \cdot Domain \in Category_{ik} \cdot Synonyms$) 或者 ($Category_{ik} \cdot Synonyms \cap Category_{jl} \cdot Synonyms \neq \Phi$) 并且

($Category_{ik} \cdot Specialization \in Category_{jl} \cdot Specialization$);

② ($Purpose_{ik} \cdot Function = Purpose_{jl} \cdot Function$) 或者 ($Purpose_{ik} \cdot Function \in Purpose_{jl} \cdot Synonyms$) 或者 ($Purpose_{jl} \cdot Function \in Purpose_{ik} \cdot Synonyms$) 或者 ($Purpose_{ik} \cdot Synonyms \cap Purpose_{jl} \cdot Synonyms \neq \Phi$) 并且 ($Purpose_{ik} \cdot Specialization \in Purpose_{jl} \cdot Specialization$)。

因此,两个 Web 服务的和是可合成的当且仅当同时满足以上定义 3、定义 4 和定义 5。

5 结 语

Web 服务以及 Web 服务合成技术有广泛的应用前景。但是,在结合语义 Web 的相关技术之前,Web 服务合成模型主要基于语法(Syntax)信息,本文讨论的核心问题是如何把 Web 服务合成的相关概念与语义 Web 技术中的本体(Ontology)相结合,从而在 Web 服务的合成技术中融入更多的语义(Semantic)信息。

Web 服务的自动合成是目前的研究热点,可合成性定义是关键之一。如何在本文讨论的问题基础上融合更

多的语义信息到服务合成本体中是今后的一个研究方向,如在 Web 服务合成模型中引入“服务合成质量”(Quality of Composition, QoC)的本体信息。

参考文献:

- 1 W3C, Web Services Architecture Requirements. <http://www.w3.org/TR/wsa-reqs/> (Accessed Mar. 25, 2007)
- 2 W3C, OWL - S: Semantic Markup for Web Services. <http://www.w3.org/Submission/OWL-S/> (Accessed Mar. 25, 2007)
- 3 Sheila A. et al. Semantic Web Services. IEEE Intelligent Systems, 2001, 16(2): 46 - 53
- 4 Berners - Lee T. Semantic Web Road Map. Internal Note, W3C. <http://www.w3.org/DesignIssues/Semantic.html> (Accessed Mar. 25, 2007)
- 5 崔俊涛. 一种在 Web Services 的语义合成中的本体建模方法. 计算机研究与发展, 2004, 41(10): 209 - 231
- 6 刘大昕, 姚从磊, 曹瀚. 基于 Web 服务的工作流系统应用集成方案. 计算机工程, 2004, 30(23): 159 - 161

(作者 E-mail: hyhay@126.com)



OCLC 推出新的 WorldCat 资源选择服务

OCLC 推出了新的 WorldCat 资源选择服务,通过将资源选择和订阅过程转移到网络上,这一服务将节省图书馆的时间和资金,使资源选择和订购过程更为流畅。

OCLC 已经与 Cornell 大学图书馆合作,在 Cornell 大学图书馆的集成选书和订购工具(Integrated Tool for Selection and Ordering at Cornell University Library, ITSOCUL)的基础上实施 WorldCat 资源选择服务。这种新服务可以使资源选择者在一个系统中看到多个资源卖主的资源条目。

WorldCat 资源选择服务将资源采集工作的中间流程(选择过程)自动化了。图书馆员仍然需要“选择”资源,但 WorldCat 资源选择服务省去了发送纸本单子和从多个卖主系统中选择的麻

烦。负责资源采集的工作人员可以将 WorldCat MARC 记录自动嵌入到图书馆集成系统中,而无需再从多个数据源中录入数据。

Cornell 大学、OCLC 和资源卖主之间的这项合作将可能会进一步推动机构技术服务过程的流畅化并节约资金。最终 WorldCat 资源选择服务将为那些依赖新资源来推进自己的工作用户带来方便。

(编译自:WorldCat Selection Service Helps Streamline Selection, Ordering. <http://www.oclc.org/news/releases/200645.htm>. [2007 - 01 - 07])

(本刊讯)