

语义 Web 和 UDDI 集成的研究*

Integrating Semantic Web into UDDI

马可方, 李陶深

MA Ke-fang, LI Tao-shen

(广西大学计算机与电子信息学院, 广西南宁 530004)

(School of Computer, Electronics and Information, Guangxi University, Nanning, Guangxi, 530004, China)

摘要:选择 OWL 来描述本体, 借鉴语义 Web 的思想, 提出将基于本体的语义和 Web 服务进行结合, 使用本体对 Web 服务进行语义描述, 对统一描述、发现和集成(Universal Description Discovery Integration, UDDI)进行扩充, 实现对 Web 服务发现技术的语义支持, 使得基于语义的 Web 服务发现过程比原有的基于服务属性的服务发现过程更为准确和有效。

关键词: Web 服务 语义 Web 本体 服务发现

中图分类号: TP393.092 **文献标识码:** A **文章编号:** 1002-7378(2006)04-0244-02

Abstract: In the combination of Ontology semantic and Web services, the web services are described in Ontology, and Universal Description Discovery Integration (UDDI) is extended. In the operation, the service discovery process based on semantic is more accurate and efficient than the current discovery mechanism which is conducted by matching of service attributes values.

Key words: Web services, semantic Web, ontology, service discovery

随着 Web 服务应用的不断增长, 用户常常要面对不同表现形式的 Web 服务, 如何准确、高效地找到所需的服务变得十分重要。传统的 Web 服务标准使用 XML 对服务进行描述, 然而 XML 最大的局限性就是采用基于语法的描述方式, 缺乏明确的语义, 即两个完全相同的 XML 描述片断在各自的上下文中可能代表完全不同的事物。Tim Berners-Lee^[1]在 XML2000 会议上首次提出语义 Web 的概念为解决这一问题提供了良好的途径。

在语义 Web 中, 不仅可以将语义添加到信息资源(如 Web 页)中, 也可以将语义添加到 Web 服务中, 这样就能共享和集成信息资源和应用程序。要在语义层次上实现信息的互操作性, 需要对信息的意思在领域里有一个一致的理解。语义 Web 引入了本体的思想, Borst^[2]给出了本体的定义。本体为某领

域提供了一个共享的、通用的理解。

目前, 可用于描述本体的置标语言主要包括 RDF (Resource Description Framework)、RDF Schema、DAML + OIL (DARPA Agent Markup Language Ontology Inference Layer)、OWL (Web Ontology Language)。OWL 已经成为 W3C 组织推荐的标准, 其设计思想是以尽可能清晰的语义描述需要发布的资源(即本体范围内的不同名词)使机器系统之间能够最大程度地理解信息的语义, 实现资源在网络结点间的开放和互操作。OWL 是 DAML + OIL 的延伸, 基本语法和功能与 DAML + OIL 接近, 而且具有更丰富的语义和关系逻辑表示功能。本文选择 OWL 来描述本体, 借鉴语义 Web 的思想, 提出将基于本体的语义和 Web 服务进行结合, 即在 Web 服务中添加必要的语义注释, 使用本体对 Web 服务进行语义描述, 对统一描述、发现和集成(Universal Description Discovery Integration, UDDI)进行扩充, 实现对 Web 服务发现技术的语义支持。这将极大地提高 Web 服务发现的查全率和查准率。

收稿日期: 2006-07-17

作者简介: 马可方(1980-), 男, 广西南宁人, 硕士研究生, 主要从事智能搜索引擎研究。

* 广西自然科学基金项目(桂科自 0640026)资助。

1 结合的思路

要想使 Web 服务发现技术向语义层面发展,必须对 Web 服务的某些协议栈层次进行修改和扩充,使其支持语义信息,从而才能达到语义发现,向更高的智能化程度发展。UDDI 层次距离 Web 服务发现比较近,又因为他们处于 Web 服务协议栈的高层,具有较多的元素,也比较容易扩充,因此,我们利用本体论相关技术对其进行一定的改造,相对于修改更底层协议栈来说所花费的精力要小。

2 OWL 与 UDDI 的结合

UDDI 注册中心使用的核心信息模型由 XML Schema 定义。XML Schema 定义了商业实体信息、服务信息、绑定信息和技术规范信息四种主要的信息类型。服务在 UDDI 结构中的表示如图 1 所示。

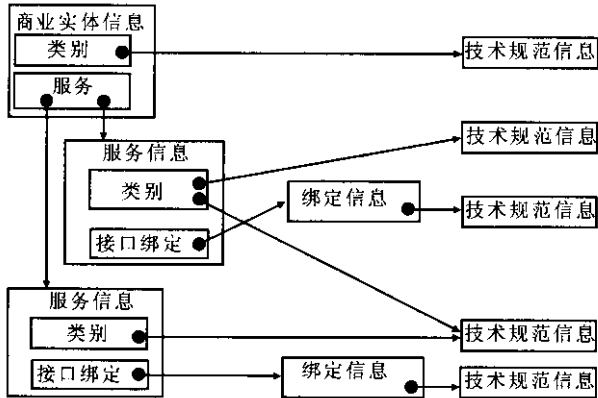


图 1 服务在 UDDI 中的表示

其中,技术规范信息的主要是作为描述一种技术规范的机制存在的。一个技术规范信息是描述 UDDI 注册中心中各种商业、服务和模板结构的一种方式^[3]。技术规范信息结构用来描述与某一规范、概念或共享设计的兼容性。技术规范信息使得不同的公司可以通过技术规范信息识别不同的规范或者在特定合作者间共享某些已经存在的规范。任何抽象的概念都可以在 UDDI 注册中心以一个技术规范信息的形式进行登记。

基于技术规范信息的上述特点和机制,在这里我们采用技术规范信息的方法对 UDDI 层进行语义方面的扩展会非常的方便,并且与现行的标准的 UDDI 注册中心不产生冲突,包含语义的 Web 服务注册信息可以方便的保存在 UDDI 注册中心,不需要对原来的架构进行大的调整,发现技术也可以通过 UDDI 注册中心找到特定的技术规范信息从而打开其中封装的语义编码。

参照文献[4]中的 DAML-S/UDDI 映射机制,采用一对一映射的方式,建立 OWL/UDDI 映射机制:我们创建 5 个技术规范信息用于装载 OWL 赋予的其他 5 个元素,分别是 `rdfs:Property`、`rdfs:subPropertyOf`、`rdfs:domain`、`rdfs:range`、`Individual` 属性用于说明个体之间或个体到数值之间的关系,子属性类相似,可以通过说明一个属性是另一个或多个属性的子属性来建立属性之间的层次关系,属性的域限定了可以应用该属性的个体的类,属性的范围在另一个方向上限定个体的取值,个体是类的实例。可以用属性将一个个体与另一个个体关联起来。在定义的 5 个非原 UDDI 的技术规范信息里,其元素结构和其他的技术规范信息保持一致。当一个新的技术规范信息被注册到 UDDI 中心后,它的类型便被归类到 UDDI 类型分类法框架中。对于每一个相应的划分,技术规范信息的引用键值都将被加进 `categoryBag` 元素中。下面给出技术规范信息装载 `rdfs:Property` 元素的具体描述:

```
< tModel authorizedName = "... " operator =
= "... " tModelKey = "AAAAAA" >
  < name > Service Property < /name >
  < description xml:lang = "en" >
    Property of service
  < /description >
  < overviewDoc >
    < description xml:lang = "en" >
      equivalence of OWL rdfs:property
    < /description >
    < overviewURL >
      http://localhost:8888/OWL/example.owl
    < /overviewURL >
  < /overviewDoc >
  < categoryBag >
    < keyedReference
      tModelKey = "uuid:0e727db0-3e14-11 d5-
98bf-002035229c64"
      keyName = "uddi-org:property"
      keyValue = "owlproperty"
    < /keyedReference >
  < /categoryBag >
< /tModel >
```

(下转第 249 页)

上执行字节码程序,这时候,安全管理器始终监测所执行的没一步操作,检查其合法性,安全服务和工具按预先的设置实现其安全功能。

3 结束语

在 Java 安全体系结构中,Java 虚拟机内置了许多安全特性,在语言层确保 Java 应用程序的安全性,而且在字节码运行前通过字节码校验器和类装载器保证代码的安全性,运行时,实现预先定制的 Java 安全服务和工具的功能,并通过安全管理器始终监控 Java 程序的运行。Java 安全系统结构的各个组成部分都能为 Java 程序提供安全性。

参考文献:

[1] TRENT J, JOCHEN L, VSEVOLOD P, et al. Security

architecture for component-based operating systems [M]. ACM SIGOPS European Workshop, 1998: 222-228.

[2] BILL VENNERS. 深入 Java 虚拟机 [M]. 曹晓刚, 蒋靖, 译. 第 2 版. 北京: 机械工业出版社, 2003: 25-72.

[3] SUN Microsystems Inc. The Java virtual machine specification [EB/OL]. [2006-06-10] 2nd Edition. <http://java.sun.com/docs/books/vmspec/>.

[4] LI GONG. Java™ 2 Platform Security Architecture [EB/OL]. [2006-06-10]. <http://java.sun.com/j2se/1.5.0/docs/guide/security/spec/security-spec.doc.html>.

[5] 幸运伟, 饶一梅, 张钧. Java 程序设计 [M]. 北京: 清华大学出版社, 2004: 3-6.

(责任编辑: 邓大玉)

(上接第 245 页)

3 结束语

本文选择 OWL 描述本体,并用它实现对 Web 服务的标准 UDDI 的扩充。该方法用本体来描述 Web 服务的语义信息,使得 Web 服务中包含了适当的语义信息,从而服务发现就可以利用服务中的语义信息,这将大大提高 Web 服务发现的查全率和查准率。下一步我们将开发更丰富的本体来描述操作、前提和结果的功能;研究服务发现算法以便更充分地利用 WSDL 和 UDDI 中的语义,其中将重点研究根据匹配相似度排列查询结果的最佳算法。

参考文献:

[1] BERNERS-LEE T, HENDLER J, LASSILA O. The Semantic Web [J]. Scientific American, 2001, 284 (5):

34-43.

[2] BORST W N. Construction of engineering ontologies for knowledge sharing and reuse [D]. PhD thesis, Twente University, 1997.

[3] 柴晓路. tModel 的用途及结构详解 [EB/OL]. 2001 [2006-07-15]. <http://www-128.ibm.com/developerworks/cn/webservices/ws-tmodel/part/index.shtml>.

[4] PAOLUCCI MASSIMO, TAKAHIRO KAWAMURA, TERRY R PAYNE, et al. Importing the Semantic Web in UDDI: Proceedings of Web Services, Ebusiness and Semantic Web Workshop [C]. London: Springer-Verlag, 2002: 225-236.

(责任编辑: 凌汉恩 邓大玉)