

基于 workflow 技术的复合语义 Web 服务框架

李湘梅^{1,2}, 孙 萍^{1,2}

(1. 同济大学电子与信息工程学院, 上海 201804; 2. 国家高性能计算机工程技术研究中心同济分中心, 上海 201804)

摘 要: 随着 Internet 和 Web 应用的快速发展, Web 应用模式及 Web 服务的异构和分散问题日益突出, 如何复合繁复而庞大的 Web 服务群成了一项繁杂的工作。该文描述了一个复合语义 Web 服务框架的设计和实现过程。利用本体模型及 workflow 技术, 该 Web 服务框架能够动态复合基本 Web 服务并在动态 Web 环境下自动执行, 提高了 Web 应用的灵活性, 并结合语义 Web 和 workflow 理论在提供智能化 Web 服务方面作了一次新的尝试。

关键词: Web 服务; 语义 Web; 本体; workflow

Composite Semantic Web Service Framework Based on Workflow Technique

LI Xiang-mei^{1,2}, SUN Ping^{1,2}

(1. School of Electronics and Information Engineering, Tongji University, Shanghai 201804;

2. National Engineering and Technology Center of High Performance Computer Tongji Branch, Shanghai 201804)

【Abstract】 The representation of Web application pattern gains a considerable momentum with the rapid development of the Internet and Web application. Unfortunately, as Web services are often autonomous and heterogeneous, how to composite them is a delicate and time-consuming task. This paper describes the design and realization of a framework through which existing Web services can be dynamically composed, and the composite services can be executed following a suitable sequence within a dynamic environment. This framework provides tools for specifying composite services through ontology and workflow theory. Consequently, the research efforts related to the work is a valuable attempt to increase the Web application pattern's flexibility.

【Key words】 Web service; semantic Web; ontology; workflow

1 概述

随着 Internet 的高速发展和 Web 应用的不断深入, Web 应用模式的表现形式日益丰富。不仅 Web 服务的数量越来越庞大, 而且运行在不同平台之上的 Web 服务大多是异构的, 面对如此庞大而繁杂的 Web 服务群, 有必要研究如何从中动态查找到用户所需的 Web 服务, 集成这些异构的 Web 应用, 以便能够实现 Web 服务的智能化查找。传统的 Web 服务模式以及各种用 XML 语法描述的 Web 服务协议缺乏良好的语义信息, 同时也缺乏异构 Web 服务协同交互的表达能力; 另一方面, 由于 Web 服务本身没有流程的概念, 简单 Web 服务往往无法满足复杂多变的实际需求, 不能适应开放的、动态的 Web 环境, 在这种情况下, 如何保证在动态环境下异构 Web 服务的自动组合、正确配置和合理调度就显得尤为重要。

2 语义 Web 基础

2.1 语义 Web 的概念及体系结构

Tim Berners-Lee 于 2000 年 12 月在 XML 2000 会议上正式提出了语义 Web 的概念和体系结构^[1], 语义 Web 是一个网, 它包含了文档或文档的一部分, 描述了事物间的明显关系, 且包含语义信息, 以利于机器的自动处理。也就是说, 语义 Web 的目标是使得 Web 上的信息具有计算机可以理解的语义, 满足智能软件代理对网络上异构和分布信息的有效访问。

Tim Berners-Lee 所提出的语义 Web 的体系结构分为 7 个层次(见图 1): 第 1 层是 Unicode 和 URL, 它是整个语义 Web

的基础, Unicode(统一编码)处理资源的编码, URL(统一资源定位符)负责定位资源; 第 2 层是 XML + 名字空间 + XML 模式, 用于表示数据的内容和结构; 第 3 层是 RDF + RDF 模式, 用于描述资源及其类型; 第 4 层是本体词汇, 用于描述各种资源之间的联系; 第 5 层是逻辑, 在下面 4 层的基础上进行逻辑推理操作; 第 6 层是验证, 根据逻辑陈述进行验证以得出结论; 第 7 层是信任, 在用户间建立信任关系。

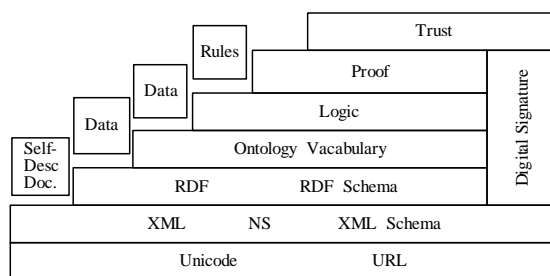


图 1 语义 Web 的层次结构

基金项目: 国家自然科学基金资助项目“面向网络服务的开放式移动计算模型、理论及方法研究”(90612006); 国家自然科学基金资助项目“网络程序层次模型及其理论”(60473094)

作者简介: 李湘梅(1971-), 女, 讲师, 主研方向: 网络计算, 并行计算; 孙 萍, 博士研究生

收稿日期: 2008-10-14 **E-mail:** lixm163@163.com

2.2 本体模型及其描述语言

本体(Ontology)^[2]最早是一个哲学概念,即存在的研究或科学。简单地说,本体相当于一般化的模式,提供了一套对特定领域知识共享的共同认识。本体的主要功能是信息交换、互操作性和系统工程等,因此很适合于描述异构的、分布的和半结构化的信息资源,如Web上的信息。本体为Web服务提供了语义,通过对概念的严格定义和概念与概念之间的关系来确定概念的精确含义,表示共同认可的、可共享的知识。因此本体是解决语义层次上Web信息共享和交换的基础和关键。

目前已有的对本体进行描述和构建的语义 Web 语言均基于 XML,包括 RDF(Resource Description Framework), RDF Schema, DAML(The DARPA Agent Markup Language)+OIL, OWL(Ontology Web Language)等。其中 DAML+OIL 和 OWL 是建立在人工智能知识表示基础之上的语言,很适合构建语义 Web。

3 workflow 应用于复合语义 Web 服务的基本概念

根据 workflow 管理联盟(WFMC)的定义^[3]: workflow 是一类能够完全或者部分自动执行的经营过程,根据一系列过程规则,文档、信息或任务能够在不同的执行者之间传递、执行。

在语义 Web 服务环境下, workflow 定义机制必须能够对各种复杂情况进行控制,使得基于 XML 的 Web 服务规范处理流程更为有效。基于 XML 以及与 Web 服务相关的业务流程管理系统标准有: ebXML BPSS, WSFL 和 XLANG 等,本系统使用 WSFL(Web Service Flow Language)作为建模语言。

4 复合语义 Web 服务框架系统结构的设计与实现

通过对本体的描述给出 Web 服务的精确语义,可以有效地实现 Web 服务的动态发现和选取,从而提高 Web 服务自动查找的准确性。从组合 Web 服务的角度来看,基于语义信息进行 Web 服务的组合能够有效地利用知识表示及推理来指导和监督服务的选取,从而确定合理的 Web 服务方案。利用 workflow 的思想来复合 Web 服务,将基本 Web 服务看作 workflow 中的一个活动,选用合适的 Web 服务工作流引擎,可以方便地确定调用顺序和工作流程。

4.1 复合语义 Web 服务框架的体系结构

本系统采用基于 XML 的 DAML+OIL 定义本体元数据集和概念集,应用本体概念解决 Web 服务查询及复合过程中存在的异构问题,由此完成语义 Web 服务的自动查找和选取。同时通过可适用于 Web 服务的工作流引擎,将选取的基本 Web 服务视为 workflow 系统中的单个活动,将其组合成恰当流程的复合 Web 服务。整个复合过程既保持了 Web 服务的语义信息,又提供了一种 Web 服务系统动态合成的方法。系统分为 Web 服务查找模块和 Web 服务复合模块 2 个部分,整体结构如图 2 所示。

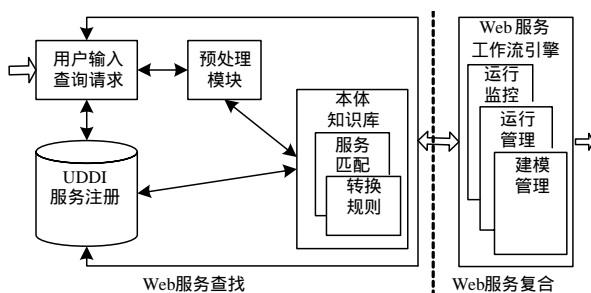


图 2 复合语义 Web 服务框架的整体结构

在 Web 服务查找模块中:

(1)用户输入查询请求:接受用户输入的查询请求。

(2)预处理模块:对用户输入的 Web 服务查询请求进行信息过滤,去除无用的信息,保留可以作为查询请求的条件和约束信息,并将该请求以标准化的形式输出。

(3)本体知识库:在本体知识库中存储有预先定义好的标准化 Web 服务集合。在接收用户查询请求后,调用其中的服务匹配模块,根据语义信息得出符合查询请求的基本 Web 服务集合。在这里采取了本体描述语言进行元数据和概念的语义描述,在协议栈里加入新的 DAML 文件描述标准服务本体。非标准服务和标准服务的转换条件和转换规则存入转换规则模块,根据标准 DAML 文件实现服务合成接口的自动匹配,由此解决部分 Web 服务查询中存在的异构问题。

(4)UDDI 服务注册容器:提供公共 UDDI 注册服务的注册中心。服务请求者及服务提供者都可与注册中心进行动态交互,用户请求可在此直接绑定已注册的服务,远程调用已配置好的服务,本体知识库也可直接与 UDDI 服务注册容器交互。注册中心提供规范的查询接口。

在基于 Web 服务的语义概念完成 Web 服务动态查找之后,进入到 Web 服务复合模块:它由一个适用于 Web 服务的工作流引擎构成,可以采用市场上成熟的开源或商业 workflow 引擎产品,如 Shark 或 JBoss 等。workflow 引擎包含了建模工具、运行管理器和运行监控模块。

(1)建模管理:即 workflow 设计器,主要用于图形化 Web 服务流程的抽象表示。

(2)运行管理:用于维护和解析复合语义 Web 流程的运行。

(3)运行监控:完成对复合 Web 服务流程的管理和监控。

4.2 Web 服务查找

在用户的查询请求经过预处理过滤掉无用信息后,由本体知识库中的服务匹配模块动态查询符合要求的 Web 服务集合。这些 Web 服务集合有可能是异构的,通过转换规则协议栈中存入的 DAML 文件的映射实现非标准服务与标准服务接口之间的转换,实现服务接口的匹配,从而解决 Web 服务异构的问题。

由于用户输入的查询请求为功能描述信息,必须根据其语义信息得出相符的同义词语、父概念词语以及子概念词语等。例如“西红柿”与“番茄”为同义词,那么两者所描述的服务都应该属于查询 Web 服务的结果集合,从用户查询条件的泛化来实现动态查找。

例如用户输入的查询请求为 a, a 与 A 为同义词, a 又为 B 的子类,与 a 相关的功能描述 Web 服务为 Service 1 和 Service 2,与 A 相关的 Web 服务为 Service 3,与 B 相关的服务为 Service 4 和 Service 5,那么在查询 a 所涉及的 Web 服务时,就可以由本体知识库中对 a 本体的描述得到 Service 1~Service 5 的查询结果集合。其中, a 本体实例在本体知识库中的 DAML 描述如下:

```
<Daml:DatatypeProperty rdf:about="a">
  <Daml:SamePropertyAs>
  <comment>for samePropertyAs ( P, R ), read P is an equivalent
property to R
  </comment>
  <Daml:Class rdf:about="A">
  </Daml:SamePropertyAs>
```

```

<Daml:subPropertyOf>
  <Daml:Class rdf:about="B"/>
</Daml:subPropertyOf>
<rdfs:domain>
  <Daml:Class rdf:about="#Service 1"/>
  <Daml:Class rdf:about="#Service 2"/>
</rdfs:domain>
</Daml:DatatypeProperty>

```

另外，本体知识库中的规则转换模块解决了一部分 Web 服务异构问题。在协议中存入 DAML 转换文件实现了标准服务和非标准服务的映射。DAML 转换文件实例如下：

```

<Daml:Class rdf:about="P">
  <Daml:SameClassAs>
    <comment>for sameClassAs ( X, Y ), read X is an equivalent
class to Y of OIL Equivalent.
  </comment>
  < rdf:about="R"/>
</Daml:SameClassAs>
</Daml:Class>

```

4.3 Web 服务复合

复合 Web 服务则是根据基本 Web 服务动态组合完成某一特定功能的 Web 服务，通过 workflow 机制的定义，它有单一的接口和调用方式。

通过对本体知识库的语义查询会动态检索出符合条件的一系列候选 Web 服务。而在 workflow 机制中，为其提供更高灵活性的是动态定位机制，它允许 workflow 引擎完全根据调用流程应用的一套规则来选择服务提供者，为处理 Web 服务提供了一种新的思路：动态联合并集成松散结合的应用程序组件。图 3 为一个简单的由 workflow 机制定义的 Web 服务流程实例。订货商定购货物，如果资金不足，则取消定购计划，否则将定购提交给生产商，生产商根据库存情况决定生产新的货物

或者用现有货物。根据 workflow 建模理论，用有向图来表示 Web 服务流，每个方框代表一个活动，即基本语义 Web 服务，如 Plan order 表示提供订货计划单的 Web 服务等。每条带箭头的实线代表从一个活动转到另一个活动的控制流程，箭头指明流程进展的方向。

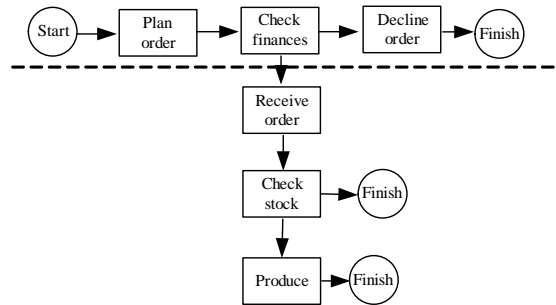


图 3 订货过程的复合 Web 服务流程

5 结束语

本文利用语义 Web 的相关理论，把 workflow 技术引入语义 Web 服务的协同复合进行系统建模和数据分析，构造了一个基于 workflow 技术的复合语义 Web 的系统框架，提高了 Web 应用的灵活性，还结合语义 Web 和 workflow 理论在提供智能化 Web 服务方面进行了一次新的尝试。

参考文献

- [1] Lee T B. Semantic Web Road Map[EB/OL]. (1998-01-21). www.w3.org/DesignIssues/Semantic.html.
- [2] Gruber T. A Translation Approach to Portable Ontology Specifications[J]. Knowledge Acquisition, 1993, 5(2): 199.
- [3] Luo Haibin, Fan Yushun, Wu Cheng. Overview of Workflow Technology[J]. Journal of Software, 2000, 11(7): 899-907.

(上接第 64 页)

在下一个实验中，以随机的批量方式更新存在的 10^4 个对象。更新数据集的比例在 5%~40% 之间变化，实验结果如图 2 所示。

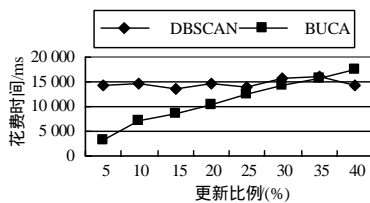


图 2 DBSCAN 和 BUCA 的性能

从图 2 中可以看出，BUCA 的运行效率随着更新对象的增加而降低，其运行时间与对象的数量大致成正比。如果更新对象的比例大于一个极限值(在本文的结果中，该值近似为 35%)，BUCA 的效率会比 DBSCAN 差，原因在于更新的对象越多，越多的相关对象需要得到检查。因此，当更新对象的比例低于 10% 时，BUCA 的优势是很明显的，这种情况在实际运用中也是最常见的。

4 结束语

在数据挖掘中，增量算法能提高聚类效率。典型的情况

是：收集更新之后以批量的方式周期性地应用到数据仓库中，而这会影响聚类结果。为仅考虑旧的聚类和那些每天插入或删除的对象，只执行这些增量的更新，而不是将聚类算法应用于更新数据库，这是种理想的做法。

本文给出了 BUCA 和 DBSCAN 的等价分析，与 DBSCAN 相比，该算法能更有效地解决增量聚类问题，进一步改进利用指数提高其性能，运用空间关联得到聚类。

参考文献

- [1] Ester M, Kriegel H P, Sander J, et al. A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise[C]//Proc. of the 2nd Int'l Conf. on Knowledge Discovering in Databases and Data Mining. Massachusetts, USA: AAAI Press, 1996.
- [2] Martin E, Hans P, Sander K J. Incremental Clustering for Mining in a Data Warehousing Environment[C]//Proc. of the 24th Int'l Conf. on VLDB. New York, USA: Morgan Kaufmann Publishers Inc., 1998.
- [3] Zhou Shuigeng. FDBSCAN: A Fast DBSCAN Algorithm[J]. Journal of Software, 2000, 11(6): 735-744.