

# 基于 Web Services 和本体的信息集成框架

张在东, 盛步云

(武汉理工大学 计算机科学与技术学院, 湖北 武汉 430070)

**摘要:**针对信息集成应用的要求,利用信息集成协议栈的层次关系,提出了一个基于 Web Services 和本体的信息集成框架。详细阐述了框架的体系结构、模块功能,介绍了其特点和关键技术。该框架克服了传统跨平台技术及数据共享技术在信息集成中的局限,实现了信息的动态集成。

**关键词:** Web Services; 本体; 信息集成

**中图分类号:** TP391

**文献标识码:** A

**文章编号:** 1673-629X(2009)03-0134-03

## Information Integration Framework Based on Web Services and Ontology

ZHANG Zai-dong, SHENG Bu-yun

(School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070, China)

**Abstract:** According the requirements of application information integration, using the layer relations of information integration, give out an information integration frame which based on Web services and ontology. In this paper, elaborate the architecture and function modules of the framework and introduce its characteristics and key technologies. This framework overcome the limitations in traditional cross-platform technology and data sharing technology of the information integration and realize the dynamic integration of information.

**Key words:** Web services; ontology; information integration

### 0 引言

信息集成的主要作用是在企业应用集成中处理和管理分布的信息资源,为上层应用提供一致接口和高质量的可用信息。它在企业应用集成中发挥着越来越重要的作用。

对于分布式异构信息集成的问题,目前有两种解决方案:一种是利用数据仓库等将异构信息源上的数据实现物理集成,形成具有统一模式的信息源,用户直接对这个统一信息源进行操作。这种方法造成重复存储大量数据,且数据更新麻烦。另一种方法是保留数据的分布式存储,增加了一个虚拟的集成视图,用户可以通过视图与数据的映射关系直接从信息源获取数据。这种方法保证了共享信息的及时更新,但由于创建和修改虚拟视图操作复杂,信息源的频繁变化会导致虚拟视图变得难以维护。显然以上两种方法都不能满足虚拟企业信息系统对动态性和可扩展性的要求。

为了满足信息集成系统对动态性和可扩展性的要求,文中提出了一种基于 Web Services 和本体的信息集成框架。该框架完全屏蔽了平台和信息源的差异,可达到异质对象的透明访问,同时具有动态性和可扩展性,克服了传统分布式对象技术和异构信息源共享方案中的局限。

### 1 信息集成的协议架构

从技术角度看,信息集成经历了两大阶段:第一阶段以 TCP/IP 为标志,实现的是内容上的低成本协同,通过 TCP/IP 协议,各种计算机系统屏蔽了硬件,实现了不同操作系统的互联。第二阶段以 XML 为标志,实现的是应用上的低成本协同,通过 XML,各种应用系统可以实现在语法层次上的互联。不同的发展阶段和应用,实现了不同层次上的信息集成。文献[1]给出了如图 1 所示的信息集成的层次。该层次图类似 ISO7 层网络协议栈,称为信息集成协议栈。

协议栈的第七层是会话层,通过对话和互操作协议,定义了系统之间的消息互操作的规范,它可以是简单的请求/响应模式,也可能是复杂的招投标方式;第六层是通信活动层,定义了标准的、与应用无关的方

收稿日期:2008-07-02

基金项目:湖北省自然科学基金资助项目(2007ABA218)

作者简介:张在东(1979-),男,山西忻州人,硕士研究生,研究方向为计算机应用技术、信息管理;盛步云,博士,博士研究生导师,教授,研究方向为计算机集成制造。

法,在系统之间传递基于语义的消息;第五层是内容层,通过谓词逻辑或附加的限制,描述消息的实际内容;第四层是本体层,适用于某个特定问题域的共享词库,它定义了相关的术语、描述及其关系;第三层是语法层,它定义了内容的表述规则,是所有语义层的基础。语法层可选用的技术包括 XML 和 Web Services 等。XML 是一种元标记语言,根据预定义的语法,人和程序都能理解。由 XML 表示的简单对象访问协议 (Simple Object Access Protocol, SOAP) 是一种轻量级的对象访问协议,SOAP 是基于 HTTP、SMTP 等网络协议,它可以轻松地穿越防火墙。信息集成协议栈的最下面两层分别对应 ISO 七层协议的第七层与第六层。

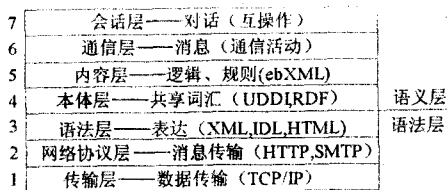


图1 信息集成协议栈

## 2 信息集成框架

因为 Web Services 的核心支撑技术是: XML, SOAP, WSDL, UDDI 以及扩展的协议栈,所以 Web Services 可提供网络服务,可跨语言、跨平台、跨操作系统并能容易穿越防火墙<sup>[2]</sup>。

目前基于本体的信息集成方法主要有三种<sup>[3]</sup>: 单本体方法、多本体方法和混合本体方法。单本体方法适用于待集成所有信息源具有几乎同样的视图。如果某个信息源具有不同视图,寻找最小的本体承诺 (ontology commitment) 就变得很困难了,所以单本体方法对于影响概念化的信息源是敏感的。多本体方法由于缺乏共同的词汇表,不同源本体的比较极为困难。因为不同的源本体可能由不同的组织建立起来,所用的模式很可能是不同的,甚至相差很大,所以多本体方法也不适用于分布式网络环境下的信息集成。混合本体克服了单本体和多本体方法的缺陷,在不同的信息源上建立各自的局部本体,避免了局部结构的改变对全局的影响。在各局部本体上建立共享的全局本体,通过全局本体实现局部本体之间的互操作。文中采用混合本体的方法来解决语义异构问题,采用 Web Services 技术解决语法异构的问题。结合信息集成协议栈提出了信息集成架构的层次结构,如图2所示。

### 2.1 框架的体系结构

图2为框架系统的框图,共分为三层,分别为信息

源层、信息集成服务层和用户接口层,功能如下:

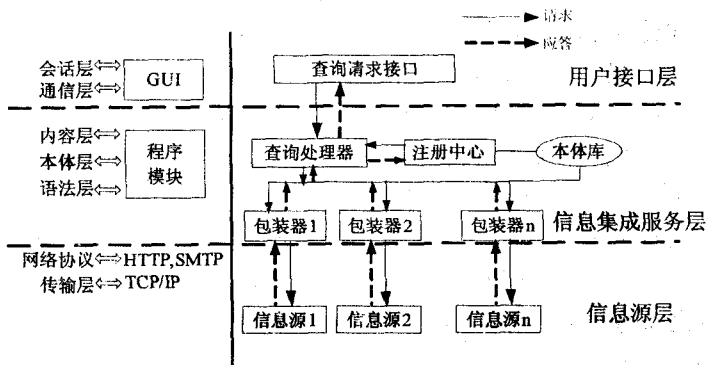


图2 信息集成架构层次

(1)信息源层。信息源层是系统的最底层,同时也是整个系统的基础,主要包括各个分布、异构信息源。底层信息的传输依靠信息集成协议栈的传输层和网络协议进行传输,基于 HTTP 的协议可轻易穿越防火墙,实现跨系统信息传递。

(2)信息集成服务层。该层是异构信息源实现共享的核心,对应于信息集成协议栈的内容层、本体层和语法层。负责从上一层接收查询请求。并根据已有的服务注册信息将请求分解成针对不同信息源的子请求,产生查询执行方案,按照执行方案调用服务。并将调用结果以统一的模式传递给上一层。该层四个基本组成部分的主要功能如下:

①本体库:包括全局本体、局部本体、全局本体和局部本体以及局部本体和对应信息源之间的映射。包含所有信息源的通用语义模型。该语义模型规定了待集成各信息源的数据对象的模式和数据的语义,从用户查询处理、全局数据视图的定义、各信息源局部数据视图的定义、信息源描述的生成到基于领域知识的信息检索与匹配,可以说本集成系统的各个环节都要以本体库中的本体为指导,是实现语义查询的关键。

②查询处理器:查询处理器是本系统的核心,查询处理器包含查询处理、服务发现与发布、服务调用、结果处理等模块,查询处理器结构图如图3所示,其中查询处理模块根据全局本体框架信息,分析用户查询接口发来的查询条件包含的概念及其依赖关系,得到相关本体的属性信息,并根据全局本体和局部本体的映射关系,把全局查询分解为多个带语义信息的子查询,传递给服务发现与发布模块;在接收到返回结果时,把返回结果以统一的模式返回给用户接口层。

服务发现与发布模块把 UDDI 注册中心中的所有服务提供的语义信息与查询处理模块得到的语义条件进行语义匹配(匹配的语义包括服务的输入条件和输出信息的语义模型),根据语义匹配算法得出匹配结果集,再从语义匹配结果集中按匹配度的高低进行排序,

然后使用某种选择策略选择合适的服务;在服务发布时,把包装器送来的服务参照局部本体的语义进行描述,然后把服务注册到服务注册中心。

服务调用模块根据选取的各个服务所需的查询输入要求进行查询重写即把用户的查询条件传递给服务的输入条件,并通过服务绑定进行服务调用,完成所需信息的抽取工作。

由于一个查询可能涉及到多个信息源,各个信息源之间的模式都不尽相同,所以抽取的结果也可能有多种模式的 XML 文档,因此,查询处理器还需要结果处理模块对这些结果集数据文档进行处理,生成一个完整的查询结果。

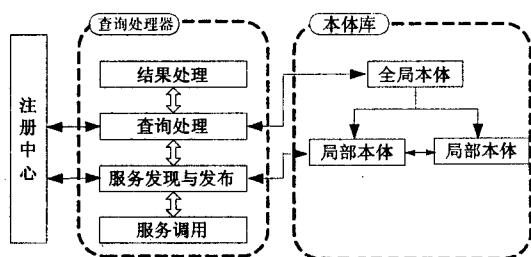


图 3 查询处理器结构与运行机制

③服务注册中心:根据包装器提交的服务描述对服务进行注册;负责将查询处理器传递过来的查询请求与具体服务对应。如果查询请求与注册的服务相匹配,则将提供有关服务细节的服务接口信息以 WSDL 格式返回给查询处理器。如果没有相匹配的信息,则返回无相关服务的说明。

服务注册中心在逻辑上起到了全局统一视图的作用。服务注册中心将异构数据源上的数据组织起来,查询处理器根据服务注册中心提供的信息将针对多个异构信息源的查询分解映射到具体的数据源上。如果有对应的服务,就发送 SOAP 请求消息进行调用,如果没有对应的消息,则返回请求无法执行的说明。但是应该注意到,服务注册中心并没有创建和维护全局视图,服务是由信息源提供,可以任意增添或删除,而不需要进行任何调整。

④包装器:当服务发布时,负责将要发布的数据资源以局部本体标准进行统一并描述封装为服务;当服务被调用时,首先会确定是哪一个服务被调用,然后从服务调用请求中取出服务输入参数,并根据数据源内字段与本体的映射关系将输入参数转换为系统内部使用的数据形式,然后执行服务中的具体操作,并将操作结果传递给服务调用模块。

(3)用户接口层是系统的最顶层,和信息集成协议栈的会话层和通信层相对应。为外部用户或其他系统提供查询访问接口,向用户展现查询结果或者将结果

传递给其他系统。

## 2.2 框架的查询处理过程

在上述体系结构中,用户查询的主要过程如下:

(1)对用户提交的查询条件由查询请求接口进行预处理,得到合法的查询条件;

(2)首先,由查询处理模块对查询请求接口发来的查询条件进行全局查询处理,根据全局本体生成全局的查询视图,分析查询条件包含的概念及其依赖关系,得到相关本体的属性信息。其次,根据全局本体和局部本体的映射关系,生成局部查询视图,把全局查询分解为多个带语义信息的子查询,传递给服务发现与发布模块;

(3)服务发现与发布模块把查询处理模块发来的带语义信息的子查询与注册中心中的所有注册的 Web 服务的语义信息进行语义匹配,根据语义匹配算法得出匹配结果集,然后使用某种选择策略选择合适的服务,把所选择的服务提交各服务调用模块;

(4)服务调用模块通过包装器执行被选择了的服务,从异构信息源得到所需的数据,将结果返回给结果处理模块;

(5)结果处理模块对得到的结果进行合并,并将合并后的结果以统一的模式作为查询结果返回给用户。

## 2.3 框架的关键技术

实现基于 Web Services 和本体的信息集成涉及多方面的技术规范,其中包括:

(1)XML:它是基于 Web Services 的信息集成的技术基础,也是其他关键技术的基础,可以说是新一代 Internet 应用的基石;

(2)SOAP 协议:它以 XML 的形式提供了一个简单、轻量级的用于分布式环境中交换结构化信息的机制。采用 XML 进行编码,提供了跨越防火墙的能力;

(3)UDDI:它提供了通过程序可以注册、发现商业实体及其 Web 服务的机制,是可以访问的 Web 服务注册中心;

(4)WSDL:它提供了用 XML 描述 Web 服务实现细节的一套规范,使程序能自动识别 Web 服务;

(5)本体构建:全局本体、局部本体及其映射的构建是实现语义集成的基础<sup>[4]</sup>,它是计算机能够识别语义信息,增强了不同系统之间的语义互操作能力;

(6)服务发现:本框架实现语义互操作的基础,解决信息集成中的“数据过剩”和“知识贫乏”的关键。

## 2.4 框架的特点

本框架具有如下的特点:

(1)整个框架基于本体技术,使用混和本体的方

(下转第 140 页)

降低系统性能。至于系统对通信质量的要求,可交由应用层来完成。

该远程控制系统同以太网接口共用一块控制芯片,实现 LED 警示灯的远程控制。系统中,PC 机为控制端,被控端为嵌入本以太网接入模块的应用系统。本设计采用类似 TFTP 机制实现控制端与被控端的通信连接。建立一个通信连接时,被控端一直监听某一固定端口,当监听到来自该端口 UDP 连接请求时,主动打开连接,然后腾出其监听的端口供其它连接使用,这一过程同 TFTP 的连接过程无异。在建立连接后,开始用户认证过程,该过程采用私有的通信机制。认证通过后,控制端才能对被控制端进行控制和状态监控。远程控制系统通信流程图如图 5 所示。

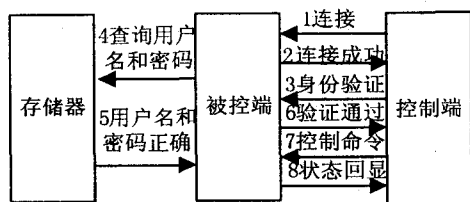


图 5 远程控制系统通信流程图

#### 4 结束语

在单片机内实现 TCP/IP 协议时,应从四个方面

(上接第 136 页)

法,为分布式异构信息源的透明访问提供了统一标准,为信息的动态集成奠定了基础;

(2)整个框架基于 XML 技术,XML 是文本形式而非二进制形式,从而提供了跨应用平台的能力;

(3)消息传递通过 SOAP 实现,SOAP 是一种基于 XML 的轻量级对象访问协议,它使用 HTTP、SMTP 等作为传输协议,从而可以跨越防火墙,使跨系统的信息集成为可能;

(4)Web Services 提供完整的封装性,可以保留遗产系统,实现渐进式开发,从而降低实施跨企业信息集成的风险和成本;

(5)Web Services 提供统一的界面,可以在第三方机构注册,可以被潜在的商业伙伴发现<sup>[5]</sup>。在任何地点,使用任何设备,通过任何方式都可以访问 Web 服务提供的服务,从而提高了跨企业集成的效用,实现“即插即用”的要求。

#### 3 结束语

文中在分析了标准的信息集成协议栈的基础上,结合本体技术和 Web Services 技术提出了一个自底向

考虑其是否合理:是否提供易用的底层硬件 API,即与硬件平台的无关性;占用的系统资源是否在可接受范围内;对于应用系统的支持程度;是否有裁减优化的空间。

文中以上述要求为标准,结合系统的硬件特点,设计实现的嵌入式 Internet 系统,成本低,易于应用。设计实现的精简 TCP/IP 协议栈简单,适用于无操作系统支持的嵌入式设备,并且易于扩充。该系统经过测试验证,设计正确,工作正常,为今后更进一步的应用奠定了基础。

#### 参考文献:

- [1] 肖洪详,邵彭飞,才娟.基于 CP2200 的嵌入式以太网接口设计[J].微计算机信息,2007(17):33-35.
- [2] Silicon Laboratories Inc. CP2200/1 Complete Datasheet[M]. USA: Silicon Laboratories Inc, 2006.
- [3] 王晓菊,潘琢金,杜睿.嵌入式网络接口模块设计及协议实现[J].沈阳航空工业学院学报,2007(1):49-52.
- [4] 刘伟,崔永峰,吴相林. TCP/IP 协议在远程单片机控制系统的设计[J].微计算机信息,2007(3):93-95.
- [5] Forouzan B A, Fegan S C. TCP/IP 协议族[M]. 谢希仁,等译.北京:清华大学出版社,2006.

上的信息集成框架。文中结合信息集成协议栈分析了框架的体系结构,详细阐述了框架的层次结构、模块功能。该框架解决了传统信息集成不能解决的语义异构问题,实现了信息的透明访问和原有系统上的信息集成,满足信息集成对动态性、平台无关性和可扩展性的要求,为进一步研究奠定了基础。

#### 参考文献:

- [1] Burg B. Agents in the world of active Web - services[EB/OL]. 2001-11. www. hpl. hp. com /org/ stl/ maas/docs/ HPL-2001-295. pdf.
- [2] 顾宁,刘家茂,柴晓路. Web Services 原理与研发实践[M]. 北京:机械工业出版社,2006.
- [3] Wache H, Ogele V, Visser T, et al. Ontology - based integration of information - a survey of existing approaches[C]//IJ-CAI-01 Workshop: Ontologies and Information Sharing. [s. l.]: [s. n.], 2001: 108-117.
- [4] Huang Ning, Diao ShiHan. Ontology - based enterprise knowledge integration[J]. Robotics and Computer - Integrated Manufacturing, 2008, 24(4): 562-571.
- [5] 周航滨,夏安邦,张长昊. 基于 Web 服务的跨企业信息集成框架[J]. 计算机集成制造系统, 2003, 9(1): 1-5.