

## 运用 Web Service 技术构建专家系统之间的交互

花 蕾<sup>1</sup>, 耿国华<sup>1</sup>, 温 超<sup>1</sup>, 雒 勇<sup>2</sup>

(1. 西北大学 计算机科学系, 陕西 西安 710069;

2. 航空计算技术研究所, 陕西 西安 710068)

**摘 要:** 专家系统解决问题的范围常常受到知识领域狭窄的限制。Web Service 实现了“基于 Web 无缝集成”的目标, 可以运用 Web Service 技术来实现专家系统之间的交互, 弥补专家系统知识不足的问题。文中运用 Web Service 提供的技术构建网络上专家系统之间的交互, 并且对这种交互的可行性进行了分析研究, 这就不仅为解决专家系统知识不足的问题提供了方法和技术, 而且进一步构建了一个模型, 使得网络上专家系统互相协作, 共同解决一个领域更广的问题。

**关键词:** 专家系统; Web Service; 通用发现、说明、集成; 简单对象访问协议

**中图分类号:** TP311.52

**文献标识码:** A

**文章编号:** 1005-3751(2006)01-0037-04

Using Web Service Technology to Design Communications  
of Expert SystemsHUA Lei<sup>1</sup>, GENG Guo-hua<sup>1</sup>, WEN Cao<sup>1</sup>, LUO Yong<sup>2</sup>

(1. Dept. of Computer Science, Northwest University, Xi'an 710069, China;

2. Aeronautics Computing Technique Research Institute, Xi'an 710068, China)

**Abstract:** The range that the expert system solves the problem often gets the narrow restriction of the knowledge field. Web service has realized the goal that “integrate on the basis of Web has not been sewn”. It can use Web service technology to realize communications of the expert systems, solving the question with insufficient knowledge of expert system. This paper applies the technology that Web service offers to make the expert system to realize communications, and has analyzed feasibility of this communications. It not only finds methods to solve the narrow restriction of the knowledge field, but also design a model to make the expert system cooperate to solve a problem in a large knowledge field.

**Key words:** expert system; Web service; UDDI; SOAP

## 0 引言

随着计算机科学特别是人工智能的发展, 人们可以用计算机程序来模拟人类的思维, 可以用人类的思维策略处理信息问题, 其中专家系统技术是人工智能走向实用化研究中最显著的成就。20 世纪 80 年代起, 专家系统在全世界得到迅速发展和广泛应用。尽管专家系统已经在各个领域得到了广泛的应用, 并收到良好的效果, 但他们解决问题的范围常常受到限制, 主要是因为: 大部分的专家系统都是针对某一特定领域建立的, 专家系统解题的领域面很窄, 所以单个专家系统的应用局限性很大<sup>[1]</sup>。

Web Service 的出现, 实现了“基于 Web 无缝集成”的目标。它是一种可透过网络存取、组合多个应用程序的软件组件所构筑出的可交互环境。文中运用 Web Service 技

术来构建专家系统之间的交互。

## 1 Web Service 技术特点

Web Service 是由 URI 识别的软件系统, 并用 XML 语言定义和描述公共接口以及绑定的信息。其他软件系统能够发现它的定义, 并能通过 IP 协议传输 XML 消息, 用其定义描述的方式与其相互作用。Web Service 具有如下特点<sup>[2]</sup>:

(1) Web Service 以 WSDL (Web Service Description Language, Web 服务描述语言) 文档对外提供逻辑描述接口, 让外界了解其提供的服务以及如何使用它。

(2) 使用 SOAP (Simple Object Access Protocol, 简单对象访问协议) 封装交换的数据及进行远程调用。

(3) 提供 WSDL 和实现层之间的绑定, 以便能够让任何程序语言或开发工具提供真正的实现程序代码。

(4) 通过 UDDI (Universal Description, Discovery and Integration, 通用发现、说明和集成) 机制, 使得 Web Service 经过注册, 用户就能够轻易地找到这些服务。

收稿日期: 2005-04-06

**作者简介:** 花 蕾 (1980—), 男, 陕西西安人, 硕士研究生, 主要研究方向为信息系统与人工智能、电子商务; 耿国华, 教授, 博导, 研究方向为基于内容检索、电子商务。

(5) Web Service 的底层协议仍然是传统的 TCP/IP 系统的网络协议和 HTTP 协议。

Web Service 突破了传统的网络壁垒的束缚,为分布式系统的开发提供了简单而健壮的基础平台。将 Web Service 与专家系统结合起来,能够使得网络上分布的专家系统服务实现共享,从而能够很好地构建网络上专家系统的协作交互。

## 2 构建专家系统交互

网络上的专家系统有着各自的运行平台和实现语言,专家系统之间的交互需要被设计成完全独立于专家系统运行的平台和语言,以便提供互操作性。

### 2.1 专家系统 Web 服务注册以及查询

WSDL 完全基于 XML,它能够完整、准确地定义提供专家系统的功能,并描述如何访问这些功能。WSDL 文档可以分为抽象定义和具体描述两部分。抽象定义部分不包含任一依赖于机器或语言的元素,它以独立于平台的方式定义提供服务专家系统的功能和功能中所涉及的信息。该部分包含 3 个的内容:类型(Type)、消息(Message)、访问接口(PortTypes)。而具体描述则定义了与网络协议和消息相关的一些内容,包括绑定(Binding)和服务集合(Services)<sup>[3,4]</sup>。

在网络中定位和发布专家系统提供的服务,可以借助 UDDI;在服务中心存储可以使用的专家服务,可以通过它定位与所需功能相匹配的专家系统的 Web 服务;只有在服务中心注册过的专家系统,它的服务才可以被找到。UDDI 作为发布和发现服务的中心,在专家系统交互过程中起到了一个桥梁的作用。UDDI 好比一个公共注册表,以一种结构化的方式保存每个专家系统的信息,并且它将相关的 Web 服务进行分类,以方便查找,图 1 给出了 UDDI 注册信息模型<sup>[3]</sup>。

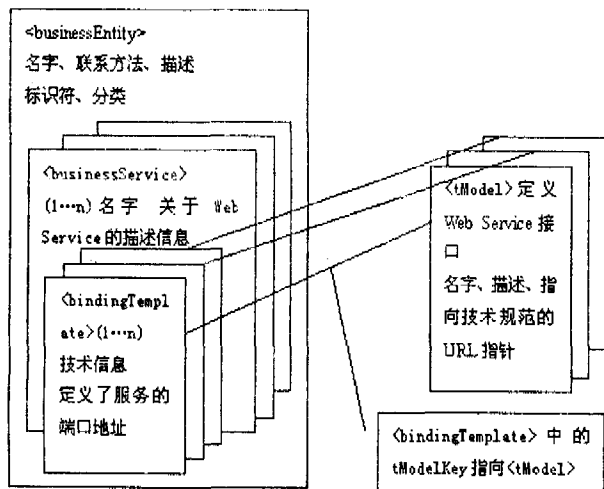


图 1 UDDI 注册的信息模型

通过 UDDI,可以发布和发现某个专家系统服务信息。UDDI 提供了查询接口和发布接口。专家系统将所部署的服务发布在 Web 上,通过 WSDL 来描述给定的专家系统

提供的功能。WSDL 完全基于 XML,它能够完整、准确地定义专家系统的功能,并描述如何访问这些功能。

发布服务的过程如图 2 所示,专家系统可以通过接口将它可以提供的服务信息发布到 UDDI 服务中心上。提供服务的专家系统主要是定义要提供给他人的专家服务,生成接口文件(WSDL 文件),并把这个服务接口和其它一些相关信息(这些信息都是 XML 文件格式的)注册到 UDDI 服务中心。并且可以获得该注册中心查找功能的描述。

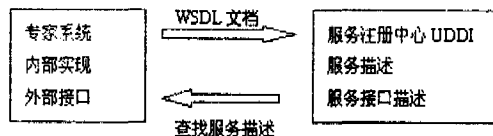


图 2 发布服务

查询服务的过程如图 3 所示,当某个专家系统需要利用网络上其它专家系统提供的服务时,这个专家系统可以与 UDDI 服务中心交互,通过注册中心的查询接口找到提供服务的专家系统的信息,这些信息包括服务的地址和服务的接口。再利用这些信息将自己的应用绑定到想要的服务上来使用这个专家系统提供的 Web 服务。

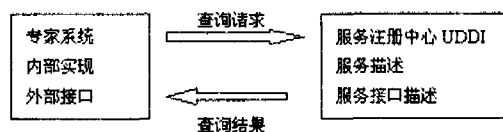


图 3 查询服务

### 2.2 Web 接口模块

由于网络上各个专家系统运行环境不同和实现语言的差异,故应该针对具体的专家系统,设计其 Web 接口。正是由于 SOAP 提供了一种与具体实现无关的对象访问能力,所以它可以将现有的、用不同的技术实现的专家系统互联起来,提供相互访问和数据交流和操作。但这种系统间的耦合又是一种很松散的耦合,它并不要对原有系统进行修改,不会影响原有系统的功能,它只需要在原有系统的基础上加上一个 Web 接口模块即可,这可以算是对原有系统的一个扩展。图 4 是专家系统 Web 接口模块的示意图。

Web 接口模块运行在专家系统平台上,由输入、输出接口、解析器以及 SOAP 引擎组成。其中,各部分功能如下:

- \* 解析器:由于网络上不同的专家系统要通过 SOAP 协议,实现相互访问,就必须将它们的请求信息用 XML 打包;在收到回复后,将包含回复信息的 XML 文档解包。这个过程是个将请求或回复信息用 XML 打包、解包的过程。

- \* 输入接口:解析器将收到的请求信息以及收到的回复信息,通过输入接口传给专家系统。

- \* 输出接口:专家系统将请求信息以及应答信息,通过输出接口传给解析器。

- \* SOAP 引擎:将包含请求或回复信息的 XML 文档包装成 SOAP 消息,并将这个 SOAP 消息通过 HTTP 协议

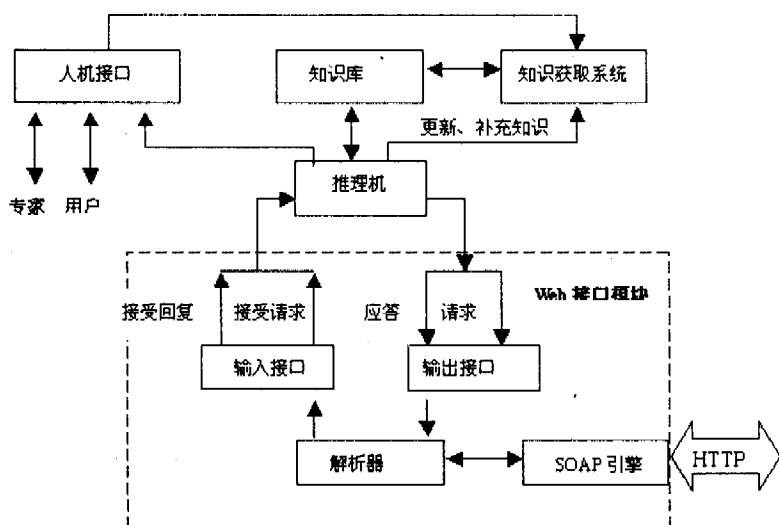


图 4 专家系统 Web 接口模块

发送;负责接受网络上传来的 SOAP 消息。

借助于 SOAP,各种不同的系统可以互相协作,形成一个更为强大的大系统。SOAP 是基于 XML 的信息交换协议,该协议屏蔽了不同环境的差异,并且能够依据底层传输协议和交互信息动态扩展,从而可以实现异构环境中多个专家系统间各种复杂的交互模式。SOAP 协议包含了封装、编码规则、远程过程调用和绑定等四部分的内容。在多专家系统的交互中,封装定义了一个框架,该框架包含传送信息、信息的发送者和接收者的标识。编码规则描述了交互信息中的数据类型,以及数据类型的 XML 描述。远程过程调用包含了专家系统之间相互调用或响应的基本信息。SOAP 消息可以使用不同的底层协议进行交换,绑定描述了如何通过底层协议在网络上的专家系统之间传送信息。

Web 接口将专家系统的请求或应答信息转换成 XML 文档,然后生成 SOAP 消息,再通过网络发送这个消息。并且能够接收通过网络传来的 SOAP 消息,将其解析成规定格式的 XML 文档,解析器分析这个 XML 文档,通过输入接口,将信息以专家系统的默认输入形式传给专家系统。

### 2.3 专家系统交互

利用专家系统的 Web 接口,专家系统可以和网络上其他专家系统进行交互。图 5 给出了一个专家系统交互模型。

利用专家系统的 Web 接口,专家系统可以请求网络上其他专家系统协助。如图 5 所示,专家系统 A 可以利用 Web 接口模块与位于网络上的 UDDI 服务中心交互,通过发送 SOAP 消息与 UDDI 服务中心交互。根据自己需要服务的类别查找 UDDI 服务中心,若符合条件的服务被发现,则定位这个服务,得到它的服务地址和服务接口。然后绑定这个 Web 服务。假如专家系统 B 被请求服务,两个

系统之间可以通过发送 SOAP 消息进行交互,专家系统 A 根据系统 B 的接口文件(WSDL 文件)生成相应的请求调用,以及参数的传递。专家系统 B 处理后,将回复的内容再通过 SOAP 消息返回给系统 A。

当某个专家系统向网络上其他专家系统请求时,将请求的内容,通过输出接口传给解析器;解析器分析这个请求,用 XML 打包,形成具有请求信息内容、标准格式的 XML 文档,然后将这个 XML 文档传给 SOAP 引擎;SOAP 引擎根据 XML 文档的内容,把这个 XML 文档包装成 SOAP 消息,然后将这个消息通过网络传出去。这个 SOAP 消息将会被传给被请求服务的专家系统。当专家系统接收到回复时,首先

SOAP 引擎接收回复的 SOAP 消息,然后形成包含回复信息的 XML 文档,并将这个文档交给解析器;解析器分析它,并通过输入接口,将回复内容传给本地的专家系统。

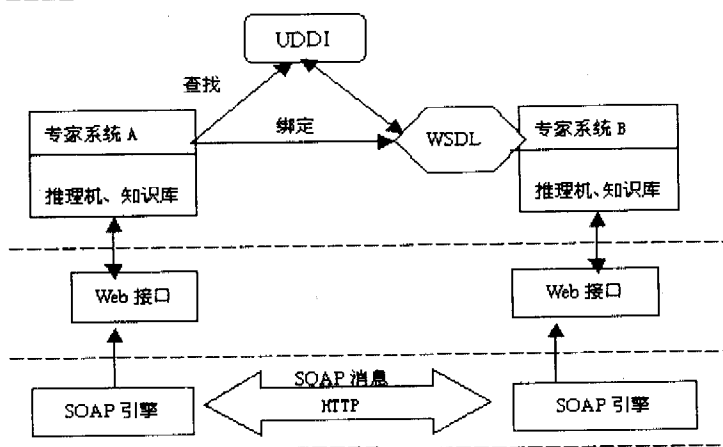


图 5 专家系统交互模型

以下例子是某个专家系统请求调用的信息模型<sup>[5]</sup>(它用 XML 语言实现):

```
<XML Description>
<messageID>消息 ID</messageID>
<messageType>消息类型</messageType>
<validateCode>验证码</validateCode>
<requestCall>
  <methodName>方法名称</methodName>
  <params>
    <param>
      <paramType>参数值</paramType>
    </param>
    <param>
      <paramType>参数值</paramType>
    </param>
  </params>
</requestCall>
</XML Description>
```

以下是专家系统接受到的回复信息模型:

```
<XML Description>
<messageID>消息 ID</messageID>
<messageType>消息类型</messageType>
<validateCode>验证码</validateCode>
<respondCall>
<methodName>方法名称</methodName>
<returnValues>
<returnValue>
<returnValueType>回复值</returnValueType>
</returnValue>
<returnValue>
<returnValueType>回复值</returnValueType>
</returnValue>
:
</returnValues>
</respondCall>
</XML Description>
```

表 1 说明

类型	描述
消息 ID	消息的标识
消息类型	4 种类型之一。分别是: 请求、应答、接受回复、接受应答
验证码	保证接受到的回复消息和请求消息的对应。专家系统可能在某个时间内收到网络上多个回复消息, 为了避免混淆, 使用验证码, 保持请求和回复之间的一一对应
方法名	专家系统的提供的调用方法名称
参数	请求方法调用时, 传入的参数值
回复值	专家系统的回复结果

可以看出, 用 XML 文档来表示请求或者回复信息, 这个信息可以被网络上的专家系统识别。这样, 可以不受专家系统运行的具体环境以及开发语言差异的影响, 只要依照接口设计标准, 开发出针对这个专家系统的 Web 接口模块, 就可以实现网络上专家系统之间的交互。

2.4 交互方式的特点

借助 Web Service 提供的一套标准和技术, 可以使得不同平台、不同编程语言的专家系统实现交互。这种专家系统间的交互方式具有以下特点:

(1) 在不同专家系统之间的传输的消息是以 XML 格

式来指定的, 这些消息都是通过 HTTP 协议发送的。

(2) SOAP 负责远程的服务请求以及应答。服务请求的编码格式在 SOAP 中是指定的, SOAP 消息本身就是以 XML 格式编码的, 它包含了对远程专家系统进行请求的方法和数据。

(3) WSDL 描述了专家系统提供服务的接口, 即请求调用的方法和接受返回的参数, 在 WSDL 文档中, 可以确定提供服务专家系统的有效 SOAP 消息格式。

(4) 使用 HTTP 协议作为底层的传输协议, 可以轻松突破不同专家系统运行环境的防火墙, 使得远程专家系统接受到 SOAP 消息。

(5) 在交互的过程中, 专家系统利用了来自网络上其他专家系统的回复内容, 能够将产生的新的知识, 通过知识获取系统即时更新、补充知识库。

3 结 论

Web Service 的出现, 为构建专家系统之间的交互提供了一个新的平台。文中给出了一个网络上专家系统的交互模型, 利用 Web Service 的技术特点实现了跨平台的交互。不同领域的专家系统之间的交互可以大大地弥补单一专家系统处理能力不充分、知识不足的弊端。

基于这种新的平台, 可以独立于软件提供者制定的标准, 构建专家系统之间的交互。只要某个专家系统经过注册, 并且开放它的服务接口, 网络上的其它专家系统就可以利用它所提供的服务。这样, 位于 Web 上的多个专家系统之间可以互相协作, 共同解决一个领域更广的问题。

参考文献:

[1] 蔡自兴, 徐光佑. 人工智能及其应用(第 3 版)[M]. 北京: 清华大学出版社, 2004.  
[2] 刘 刚, 余 晖. 利用 WSDL 和 UDDI 为公共 Web Service 建立统一接口[J]. 计算机应用研究, 2003(5): 150-152.  
[3] 孙 凯, 陈德人. 基于 UDDI 和 Web Service 的应用模型研究[J]. 计算机应用研究, 2003(5): 133-134.  
[4] 韩晓峰, 徐良贤. 基于 Web 服务的多 Agent 系统的研究[J]. 计算机仿真, 2004(1): 74-76.  
[5] Laurent S S, Johnston J, Dumbill E. Programming Web Services with XML-RPC[M]. [s.l.]: O'Reilly, 2001.

(上接第 12 页)

[3] 丁永祥, 夏 巨. Voronoi 图和 Delaunay 三角剖分的计算及应用[J]. 华中理工大学学报, 1996, 24(增刊 1): 67-72.  
[4] 张宗华, 彭 翔, 史伟强, 等. 平面域任意散乱点自动三角化的研究[J]. 工程图学学报, 2000(2): 38-45.  
[5] 谢传节, 万洪涛. 基于四叉树结构的数字地表模型快速生成算法设计[J]. 中国图像图形学报, 2002, 7(4): 394-399.  
[6] Thsesen A H. Precipitation Averages for Large Areas[J]. Monthly Weather Review, 1991, 11(39): 1082-1084.  
[7] Sibson R. Locally Equiangular Triangulations[J]. Computer

Journal, 1978, 21(3): 243-245.  
[8] 朱战立, 刘天时. 数据结构(第 2 版)[M]. 西安: 西安交通大学出版社, 1999.  
[9] Piegl L A, Richard A M. Algorithm and data structure for triangulating multiply connected polygonal domains[J]. Computers & Graphics, 1993, 14(6): 23-35.  
[10] 薛伟莲, 王志强. 可视化的 C++ 实现单调多边形三角剖分[J]. 大连海事大学学报, 2002, 28(2): 98-102.  
[11] 童爱红. 深入编程内幕——可视化的 C++ [EB/OL]. <http://202.113.13.168/netclass/computer/#>, 2002.