

构建协同商务系统的核心技术研究

王谢宁

(东北财经大学 电子商务学院, 辽宁 大连 116025)

摘要:协同商务系统是在扩展的虚拟企业的概念基础上,通过互联网实现企业内部资源的集成和企业之间流程协作的商务模式。从技术集成架构上,SOA已经成为了开发协同商务软件的核心技术和首选的架构模式,而 Web Service 是当前最适合实现 SOA 的一项技术的集合。基于 SOA 架构模式的协同商务系统体现了松散耦合、位置透明、协议独立的特点,能够支持按需应变的动态业务需求,很好地解决了企业内外应用系统集成存在的问题。

关键词:协同商务;SOA;Web 服务;企业信息系统

中图分类号:TP393

文献标识码:A

文章编号:1673-629X(2009)06-0221-04

Core Technology Research of Construction Collaborative Commerce System

WANG Xie-ning

(E-business Institute, Dongbei University of Finance and Economics, Dalian 116025, China)

Abstract:Based on the concept of extended virtual enterprise, collaborative commerce model achieve integration of internal resources and business process collaboration by Internet. From the perspective of technical integration architecture, SOA has become a core technology of the collaborative commerce software development and the preferred mode of the software structure. The Web service is the most suitable technologies collection to realize SOA. Collaborative business model based on SOA framework reflects some characteristic such as loose coupling, location transparency, protocol independence. It is a good solution of the enterprise application systems integration between inside and outside by supporting the dynamic needs of business.

Key words:collaborative commerce;SOA;Web service;enterprise information system

0 引言

协同商务是近几年的前沿热点领域,是较 B2B 更好的运作模式,被看做是第二代的电子商务。在当前网络和协同技术迅速发展的时代,企业间的合作需要通过互联网来完成协同式的商务,因此企业不仅需要考虑内部员工和部门资源的集成,还要考虑与其它企业的协作,实现流程的跨组织化,以便共同提高对顾客的响应速度,扩大企业的接触范围和接入渠道,提高整个供应链的柔性,因此,如何协调企业间众多复杂业务往来关系,是构建企业信息平台所面临的新问题。

1 协同商务的基本思想

1.1 定义与特点

协同商务是指,在全球经济一体化的背景下,利用

以 Internet 等为特征的新兴技术为实现手段,在企业的整个供应链内及跨供应链进行各种业务的合作,最终通过改变业务经营的模式与方式达到资源充分利用的目的。

现代企业需要改变原有的组织和流程,加快反应速度,明确自己的核心竞争力量,将核心流程外的业务外包或与其他企业合作开发,联合所在行业中的上下游企业,建立一条业务关系紧密、经济利益相连的供应链,使企业间形成一个优势互补、协同制造和销售的环境,共同增强市场竞争实力,协同商务使企业之间能够以 Internet 为基础,建立合作、互利的关系^[1]。协同商务的核心在于企业与企业之间以合作为基础,系统中的每个经济实体发挥自己最擅长的方面,实现强强联合,以获得最佳的竞争力。

1.2 协同商务与企业集成技术架构

协同商务战略要求企业接受一个扩展的虚拟企业的新概念,企业应用集成可以将企业中的所有的信息资源集成到一个无缝的、并列的、易于访问的企业资产中,通过建立底层结构来联系贯穿于整个企业的数据源、应用、异构系统等,完成在企业内部的 MIS、CRM、

收稿日期:2008-09-23;修回日期:2008-12-19

基金项目:大连市 2008 年度 IT 优秀教师科研成果项目(大财企 2008[281]号文件)

作者简介:王谢宁(1974-),男,博士研究生,讲师,研究方向为电子商务、Web 服务。

SCM、数据库、数据仓库以及其它重要的内部系统之间无缝地共享和交换数据。基于协同商务思想的企业信息化系统架构应用集成存在于数据层、业务逻辑层和表示层三个不同层次上,集成的数据源囊括各类 DBMS、电子邮件、HTML 文档、普通文件等结构化、半结构化和非结构化的信息集成等,系统应能够为全局应用和用户提供一个统一、透明访问一组已存在的自治、分布和异构数据源的方法。文中仅从技术层面讨论企业协同商务系统构建问题。

当前的一大热点 SOA 架构模式是面向服务的思想,相比面向对象的思想模式,它将系统各部分通过松散的耦合方式集成,它为企业内部及企业之间提供了廉价而简便的通信支持。同时,基于 SOA 体系结构的开发模式与以应用为中心的一体化应用程序具有高度开放性、可重用性等特点,因而这种开发模式一经提出就受到广泛推崇,而 Web Service 的出现极大地推动了 SOA 的发展,它现在已经成为 SOA 架构最好的实现技术,SOA 的强大和灵活性将迅速成为企业级应用系统的首选开发模式。由于它的这些技术特点与协同商务的思想极为匹配,两者相得益彰,SOA 已经成为开发协同商务软件的核心技术和首选的架构模式。

2 面向服务架构

2.1 定义与体系结构

关于面向服务架构(SOA, service oriented architecture)目前尚未有一个统一的、业界广泛接受的定义。一般认为:SOA 面向服务架构是一个组件模型,它将应用程序的不同功能单元(称为服务),通过服务间定义良好的接口和契约联系起来。接口采用中立的方式定义,独立于具体实现服务的硬件平台、操作系统和编程语言,使得构建在这样的系统中的服务可以使用统一和标准的方式进行通信。

面向服务的体系架构中共有 3 种角色,分别是服务提供者、服务消费者和服务注册中心。其中服务提供者负责服务功能的具体实现,并通过注册服务操作将其所提供的服务发布到服务注册中心,当接收到服务消费者的服务请求时,执行所请求的服务。服务消费者则是服务执行的发起者,首先需要到服务注册中心中查找符合条件的服务;然后根据服务描述信息进行服务绑定/调用,以获得需要的功能。而服务注册中心则用来提供服务提供者注册服务、提供对服务的分类和查找功能,以便服务消费者发现服务(见图 1)。

SOA 体系结构中的每个实体(组件)都扮演着服务提供者、消费者、注册中心这三种角色中的一种或多种。在这些角色之间使用了三种操作:

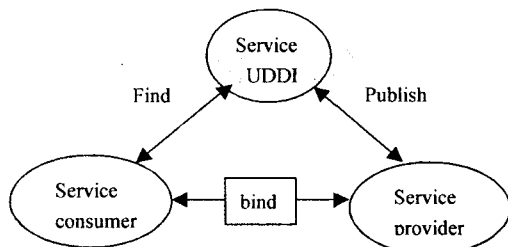


图 1 SOA 架构中的角色

(1) 发布(Publish):为使服务可访问,需要发布服务描述使消费者可以发现和调用它。

(2) 发现(Find):服务消费者定位服务,可以通过服务注册中心查找特定种类的服务。

(3) 绑定(Bind):在检索完服务之后,服务消费者根据服务描述中的信息来调用服务,使服务消费者能够真正使用服务提供者。

2.2 SOA 相关技术标准的分类

SOA 实际上是一组相关技术标准的集合,由于其复杂繁多,这里作一下分类,可以根据技术标准在 SOA 中的角色功能,将其分为三大类:服务层次上的信息交互规范、基础通信标准规范、元数据标准规范。根据各种标准规范在 SOA 体系中的角色功能,可以将 SOA 协议栈分为 7 层。从底向上,包括传输层、消息层、描述层、管理层、服务组合层、表示层,其中除了 e-BXML 和电子商务相关的技术标准(如资源注册的 e-BRS、消息表示 ebMS、外部服务资源编排的 WS-CDL 等)外,大多数在国内已经得到了相当的应用,传输层作为传统的传输协议,在 SOA 技术实现中,依然发挥着重要的作用,消息层 SOAP 已经是 Web Service 消费的消息传输载体的首选;Web Service 描述标准 WSDL,虽然在语义方面的描述还不完善,但它已经被绝大多数厂商和用户接受并使用了;在管理层的相关技术标准,目前还在发展完善,国内实际应用的还不多,但诸如常用的安全要求 WS-Security、可靠传输要求 WS-Reliability 等,已经有用户和厂家开始考虑使用;服务组合层,已经有不少的商业及开源组织,基于 BPEL 标准来开发业务流程管理软件了;表示层的标准如 JSR168 和 WSRP,主要用于 Portal 软件的开发^[2-5]。

事实上,SOA 是一种模式架构,必须要有相应的技术实现,SOA 的成功在很大程度上归功于 Web Service 标准的成熟和应用的普及为广泛地实现 SOA 架构提供了基础,如同当年的 OSI 网络七层协议一样,为网络的分层思想提供了模型,但其架构过于全面和复杂,真正在网络上广泛实现的是 TCP/IP 协议的四层结构。从目前情况来看 Web Service 无疑是当前最适合实现 SOA 的一项技术的集合。

3 Web Services 技术

3.1 概念

Web Services 是建立在开放标准和独立于平台的协议基础之上的分布计算单元。Web Service 本身由四大部分组成:XML、SOAP、WSDL、UDDI。Web Service 技术本身的兴起也正是依托这四大开放标准的发展过程和标准化历程而发展起来的。它使用 SOAP 协议在服务提供者与服务使用者之间进行通信。Web Services 通过 WSDL 协议定义服务接口,使用 UDDI 协议进行 Web Services 注册和查找。

3.2 Web Services 与 SOA

SOA 与 Web Services 是两个不同层面的问题,SOA 是一个概念,一种理念,一种模型,面向商业应用,SOA 的具体实现技术包括 CORBA、DCOM、J2EE、Web Service 等;后者则是实现模式,面向技术框架。SOA 概念并没有确切地定义服务具体如何交互,而仅仅定义了服务如何相互理解以及如何交互。其中的区别也就是定义如何执行流程的战略与如何执行流程的战术之间的区别。另一方面,Web Service 在需要交互的服务之间如何传递消息有具体的指导原则。从战术上实现 SOA 模型是通过 HTTP 传递的 SOAP 消息的 Web Service 模型。因而,从本质上讲,Web Service 是 SOA 的最佳的实现技术。这是由于 Web Service 是标准的,它保证了来自不同的 Web Service 即使运行在不同的平台上,底层的实现机理不同也可以顺利交互和共享,这是传统的技术如 CORBA、EJB 或 DCOM 等都是做不到的,更为突出的是 Web Service 的定义和实现是分开描述的,即它的松散耦合性,可以方便地修改 Web Service 内在的实现而不会对现有的系统造成破坏,这极大地促进了信息架构的灵活性。基于 Web 这种标准的成熟和应用的普及为广泛地实现 SOA 架构提供了基础。Web Services 通过 WSDL 协议定义服务接口。服务提供者可以通过 UDDI 发布自己的 Web Service,如供应商发布订单服务、客户支持服务等;服务消费者可以通过 UDDI 查找企业或者平台提供的 Web Service,如客户需要查找供应商提供的订单状态查询服务等^[6]。

从本质上来说 Web Service 是 SOA 的一种实现,但是并不是实现 SOA 的惟一形式(比如 CORBA)。但是毫无疑问,它是最成功、最流行的形式。两者之间的关系可以总结如下:

首先是基于 SOA 标准访问的独立功能实体满足了松耦合要求:在 Web Service 中所有的访问都通过 SOAP 访问进行,用 WSDL 定义的接口封装,通过 UDDI 进行目录查找,可以动态改变一个服务的提供方而

无需影响客户端的配置,外界客户端是根本不关心访问服务器端的实现。

其次,适合大数据量低频率访问符合服务大颗粒度功能:基于性能和效率平衡的要求,SOA 的服务提供的是大颗粒度的应用功能,而且跨系统边界的访问频率也不会象程序间函数调用那么频繁。通过使用 WSDL 和基于文本(Literal)的 SOAP 请求,可以实现能一次性接收处理大量数据。

最后,基于标准的文本消息传递为异构系统提供通讯机制:Web Service 所有的通讯是通过 SOAP 进行的,而 SOAP 是基于 XML 的,XML 是结构化的文本消息。从最早的 EDI 开始,文本消息也许是异构系统间通讯最好的消息格式,适用于 SOA 强调的服务对异构系统的透明性。

然而,就 SOA 思想本身而言,并不一定要局限于 Web Service 方式的实现。更应该看到的是 SOA 本身强调的是实现业务逻辑的敏捷性要求,是从业务应用角度对信息系统实现和应用的抽象。

4 基于 SOA 的协同商务系统架构

协同商务系统不仅需要企业内部各部门之间的业务、不同的业务指标和目标之间以及各种资源约束的协同有效整合,还需要企业内外资源的协同,进行一个跨企业的合作,实现一个动态的企业运行模式。

用 SOA 的指导原则来构造企业协同商务系统可称为面向服务的集成方法。面向服务的集成方法通过强制分开每个服务的使用者和该服务的提供者,增加了松耦合的关键特性,从而克服了以往脆弱的、紧耦合的应用程序集成。它将已有的相关功能转化为服务来进行集成,随着项目的进行,可重用的服务越来越多,最终达到绝大多数新的集成需求都可以通过已有的服务来完成。更重要的是,由于服务的灵活性,即使已有系统迁移到新的技术平台,甚至被替代,都不会影响到依赖于这个应用的那些应用,从而保证了集成对变化的适应能力,使得业务灵活性得到了极好的展现。这也意味着从传统规模大、投入大、周期长、见效慢、风险高转向周期短、见效快、风险低、基于标准的集成方式。

基于 Web Service 实现企业协同商务应用系统的集成是以 Web Service 接口的形式将各种应用程序和信息系统进行封装、组合和集成作为服务提供者。通过工具或手动定义 Web Services 的接口描述 WSDL 文档,并把 WSDL 文档描述的内容映射到 UDDI 数据结构中去,从而在 UDDI 注册中心对 Web Services 进行注册。无论什么平台的客户应用程序(业务)首先到 UDDI 注册中心访问 Web Services 服务的注册信息,通

过注册信息访问相应的 Web Services 的接口描述 WSDL 文档,并在本地生成代理对象。以后各业务就通过此代理对象与 Web Services 进行数据交互^[7]。如果 Web Services 内部结构和实现发生了改变,只要 WSDL 描述即接口保持不变,整个系统就不需要作任何变化。所有连入网络的服务消费者(应用程序)就可以随时调用和集成这些 Web Service。基于 Web 服务的 SOA 的关键是使用标准的服务接口和松耦合的连接,其具体实现过程如下:

(1) Web 服务提供者设计实现 Web Services,并将 Web Service 通过 SOAP 协议发布给 UDDI 注册中心进行注册,通过注册信息访问相应的 Web Services 的接口描述 WSDL 文档,并在本地生成代理对象。

(2) Web Service 请求者向 Web Service 注册中心请求特定的服务,UDDI 根据请求查询 UDDI 注册中心,并为请求者寻找满足请求的服务。

(3) Web Service 注册机构向 Web Service 请求者返回满足要求的 Web Service 描述信息,该描述信息用基于 XML 数据格式以 WSDL 形式写成,各种支持 Web Service 的机器都能阅读。

(4) 利用从 Web Service 注册机构返回的描述信息生成相应的 SOAP 消息,发送给 Web Service 提供者,以实现 Web Service 的绑定和调用。

(5) Web 服务提供者按 SOAP 消息执行相应的 Web Service,并根据服务结果返回给 Web Service 请求者。

为了获得 SOA 所带来的面向未来的长期利益和优点,面向服务的集成方法需要结合企业服务总线(Enterprise Service Bus, ESB)技术和业务流程管理方法。ESB 用了“总线”这样一种模式来连接已有应用的连接器(Connector)和适配器(Adapter),以广为接受的开放标准为基础来支持应用之间在消息、事件和服务级别上动态地互联互通^[8]。ESB 是借助于服务注册管理以及问题域相关的知识(如业务方面的一些规则等)自动实现这种联通的,不需要服务请求者和提供者介入,从而实现了服务请求者和提供者分离的技术基础,使得服务请求者不需要关心服务提供者的位置和具体实现技术,双方在保持接口不变的情况下,各自可以独立的演变。将企业服务总线作为网络上分布式服务的基础结构要素,企业服务总线的面向消息松耦合的特点为面向服务集成方法提供了实现松耦合、粗粒度服务的最佳基础和必要的业务流程驱动指导,确保使细粒度服务能够编排、组合成实际企业运行时的业务流程。

在图 2 中,基于 SOA 的系统集成结构中,给出了

一个采用面向服务集成方法进行企业协同商务应用集成的参考体系结构,很好地体现了上述的思想^[9]。通过将现有的应用系统提供 Web Service 支持,使各种原有应用系统用 WSDL 统一描述出来,作为服务的接口,统一发布给系统集成服务平台来使用,在此之上,来构建企业的各种应用。

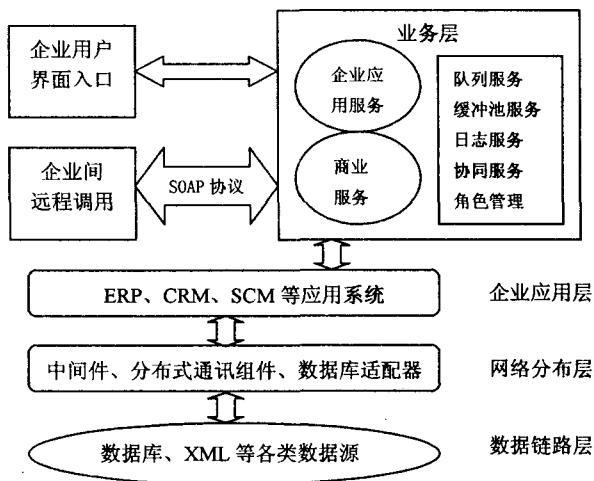


图 2 基于 SOA 的协同商务系统集成架构

5 结束语

传统的中间件技术如 DCOM、CORBA 等技术将企业内部的各种业务支撑和管理系统统一在一个平台之上,使数据运行和管理的流程更加顺畅。而 SOA 在此基础上,为协同商务提供了更为理想的集成框架,这种基于 Web Services 的方式,以 SOAP 和 XML 为通讯协议,体现了松散耦合、位置透明、协议独立的特点,能够支持随需应变的动态业务需求,很好地解决了企业内外应用系统集成存在的问题,为“信息孤岛”建立了联系,使得整个 IT 架构更加灵活,可以根据用户需求的变动进行随时调整和增删。企业内外的资源得以有效的整合,而且实现一个跨企业的合作,实现了企业的动态运行模式。

然而,从技术角度的各层应用的集成只能说是构成协同商务顺利开展的必要条件,真正的协同商务还要求实现企业内部、企业与合作伙伴、企业与客户之间的内容、流程等各方面的协作管理。所以,协同不仅仅是跨越系统的技术层面的信息交换,还蕴涵着对各协作企业之间业务流程和关系的促进。它要求对协同环境中多方位的业务流程的建模,并解决协作各方复杂的、开放式的流程和人员整合。

参考文献:

[1] 丁德臣. 企业信息化的最新趋势——协同商务综述[J]. 科

(下转第 228 页)

作为测试样本,设定神经网络的学习速度为 0.01,动量值为 0.8,目标误差为 0.01,诊断结果如表 3 所示。

表 3 基于 BP 神经网络的故障诊断结果

故障模式	训练样本		测试样本	
	故障数	诊断数	故障数	诊断数
4-K	12	12	23	23
118-K	12	12	23	23
139-K	12	12	23	23
157-K	12	12	23	23
177-K	12	12	23	23
186-K	12	12	23	23
196-K	12	12	23	23
199-K	12	12	23	23
213-K	12	12	23	23
247-K	12	12	23	23
260-K	12	12	23	23
282-K	12	12	23	23
296-K	12	12	23	23
301-K	12	12	23	23
341-K	12	12	23	23
357-K	12	12	23	23
360-K	12	12	23	23
381-K	12	12	23	23
411-K	12	12	23	23
428-K	12	12	23	23

由表 3 中可以得出,采用 BP 神经网络作为故障分类器时,总的故障诊断率为 100%。这表明:

(1)文中方法是有效的,不但可以有效降低测试矢量对之间的冗余性,得到精简测试集,减少测试成本,而且得到的测试矢量对能够使故障电路与无故障电路之间产生足够大的动态电流差异,获得较高的故障诊断率。

(2)BP 神经网络分类器方法具有对数字电路故障

准确定位能力。

5 结束语

文中利用遗传算法实现了一种动态电流精简测试集生成方法,针对电路 C432 中的部分开路故障进行测试生成。实验结果表明用该方法可以获得规模较小的测试集,包含的测试矢量对能够使故障电路和无故障电路之间动态电流产生足够大的差异,从而有利于故障的检测与定位。实验结果验证了该动态电流测试生成方法的可行性和 BP 神经网络分类器方法对数字电路故障的准确定位能力,定位率达到了 100%。

参考文献:

- [1] Aaraj N, Nazer A, Chehab A, et al. Transient current testing of dynamic CMOS circuits[C]//The 19th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT'04). Cannes, France: [s. n.], 2004: 264-271.
- [2] Wang Youren, Deng Xiaoqian, Cui Jiang, et al. Testing Cross-Talk Induced Delay Faults in Digital Circuit Based on Transient Current Analysis[J]. Wuhan University Journal of Natural Sciences, 2006, 11(2): 1145-1148.
- [3] Chehab A, Kayssi A, Ghandour A. Transient Current Testing of Gate-Oxide Shorts in CMOS[C]//The 2nd International Design and Test Workshop (IDT'07). Cairo, Egypt: IEEE press, 2007: 77-81.
- [4] 魏小芬, 邝继顺. FAN 算法在瞬态电流测试产生中的应用[J]. 同济大学学报, 2002, 30(10): 1239-1243.
- [5] 朱启建, 邝继顺, 张大方. 一种动态电流测试方法的 SPICE 模拟验证[J]. 电子学报, 2002(8): 1163-1166.
- [6] 曾晓杰, 邝继顺. 基于蚂蚁路径的 I_{DDT} 测试生成[J]. 科学技术与工程, 2007, 7(17): 4364-4368.
- [7] 王杰, 周婕, 慕晓冬. 测试数据生成技术在软件故障诊断中的应用[J]. 计算机技术与发展, 2006, 16(9): 91-92.
- [8] Min Yinghua, Zhao Zhuxing, Li Zhongcheng. IDDT testing [C]//In: Proc IEEE 6th ATS. Akita, Japan: IEEE press, 1997: 378-383.

(上接第 224 页)

技创业月刊, 2007(2): 109-112.

- [2] 汪清明. 基于 SOA 的 ERP 系统体系结构的研究[J]. 计算机应用, 2007, 27(2): 413-415.
- [3] Sprott D, Wilkes L. 理解 SOA 面向服务的体系结构[EB/OL]. 2005-10-17. <http://www.microsoft.com/china/MSDN/library/architecture/USOA.mspx?mfr=true>.
- [4] Pavlik G, Hynes D, Gral T. Next-Generation SOA Infrastructure[M]. [s. l.]: [s. n.], 2007.
- [5] Jeremy Caine, J Hardman. Design strategies for legacy system involvement in SOA solutions[EB/OL]. 2007-04. <http://www.ibm.com>.

- [6] 夏敏捷, 郑秋生. 基于 Web Service 技术的虚拟企业研究[J]. 计算机技术与发展, 2006, 16(6): 62-64.
- [7] 张大强, 殷世民, 程家兴, 等. 基于 Web Service 的电子商务体系结构[J]. 计算机技术与发展, 2006, 16(7): 22-24.
- [8] 王晓洁, 朱清新. 基于 XML/Web 服务的分布式电子商务系统的研究[J]. 计算机技术与发展, 2006, 16(3): 193-195.
- [9] 姜国辉, 赖盈汝. 企业电子化协同服务平台之研究[J]. 管理学报, 2005(2): 205-207.