

基于工作流的 Web 服务组合系统的设计与实现

王一飞, 吴素芹, 王榕

(盐城工学院 电气与信息工程学院, 江苏 盐城 224051)

摘要: 单个 Web 服务难以满足实际应用的需求, 为了解决互联网应用的集成和协作问题, 需要把独立的 Web 服务组合起来以实现复杂的业务逻辑功能。通过分析工作流与 Web 服务流程执行的相同点, 设计并实现了一个基于工作流的 Web 服务组合系统(WSCS)。该系统能在自治的、异构的、动态的环境下灵活、高效地组合 Web 服务。整个系统主要包括三个模块: 组合流程建模模块、服务选取模块、流程执行模块。

关键词: 服务组合; Web 服务; 工作流

中图分类号: TP393

文献标识码: A

Design and implementation of web services composition system based on workflow

WANG Yi Fei, WU Su Qin, WANG Rong

(Dept. of Electronic and Information Engineering, Yancheng Institute of Technology, Yancheng 224051, China)

Abstract: Single Web service just provides limited functionality, it can't provide enough functionalities for practical requirement. In order to solve the integration and collaboration problems of Internet applications, it is necessary to compose severnal Web services to implement complex business logic functionality. Though analysis workflow and Web Services, we design and implement a Web service composition system baesed on workflow, it can compose Web service in hetergenous environment. The system includes three parts: design composite flow model, service selection model and composite flow execution model.

Key words: service composition; Web services; workflow

近年来, 基于 XML 的 Web 服务技术迅速发展, 为互联网应用提供了一种共享数据的有效手段。Web 服务的高效执行方式、Web 服务与其他成熟技术的有机结合以及 Web 服务的组合是解决现实应用问题的重要技术。

Web 服务是一种自包含、自描述、模块化的程序, 它吸收了分布式计算、Grid 计算和 XML 等各种技术的优点, 解决了异构分布式计算以及代码与数据重用等问题, 具有高度的互操作性、跨平台性和松耦合性, 引起了世界范围内学术界和工业界的极大兴趣^[1-2]。

为满足 Web 服务的技术需求, W3C 等国际标准组

织制定了一系列 Web 服务规范, 如统一描述发现集成 UDDI(Universal Description, Discovery and Integration), 简单对象访问协议 SOAP(Simple Object Access Protocol), Web 服务描述语言 WSDL(Web Service Description Language) 等等, 都是用于描述、发布、发现和调用 Web 服务的, 这些规范共同构成了 Web 服务的技术体系^[3]。

Web 服务所执行的功能可以从简单的请求到复杂的商业过程中的任何事, 然而单个 Web 服务的功能有限, 难以满足实际应用中的多种多样的需求, 因此为了更加充分地利用共享的 Web 服务, 有必要将共享的

*基金项目: 国家自然科学基金项目:(项目编号: 60673175)

盐城工学院科技项目(基金项目号: XKY2007069)

Web 服务组合起来, 提供功能更为强大的服务。Web 服务组合是个非常复杂的问题^[4-7], 它涉及到 Web 服务的描述、Web 服务的发现^[8]、Web 服务的选择^[9]、Web 服务的匹配、Web 服务的调度^[10-11]、服务质量(QoS)^[12-13]等一系列问题。

本文通过参考传统工作流的相关技术, 基于 Web 服务分布式计算技术设计并实现了一个可扩展的 Web 服务组合系统(简称 WSCS), 为 Web 服务组合提供了一个运行、监控环境, 能够很好地解决 Web 服务之间的集成和协作问题。

1 总体设计

WSCS 体系结构采用面向服务的体系结构, 它所有的主要组成部件均实现为 Web 服务的形式。系统最终向用户提供图形用户界面的建模工具, 用户通过建模工具定义流程模型和模型中所涉及到的 Web 服务(可能是一个简单的 Web 服务, 也可能为一个复杂的组合服务), 并形成一个新的 Web 服务, 用于客户的应用程序中, 以满足客户的需求。

WSCS 总体设计流程如图 1 所示, 首先用户建模, 用户通过图形化界面进行建模, 将建模后的组合流程传给服务选取模块; 其次服务选取, 查找业务流程中所要用到 Web 服务, 它们可以在 UDDI 中查找, 也可以在服务社区中进行查找。在 Web 服务的标准体系中, UDDI 是一套基于 Web 的、分布式的、为 Web 服务提供信息中心的实现标准规范, 同时也包含一组使企业能够将自身 Web 服务注册, 以使客户能够发现; 再次将模型中每个业务活动同具体的业务应用进行绑定, 获得 Web 服务的本地接口, 根据用户的流程和选择的服务进行服务调用; 第四: 调用 Web 服务, 系统访问 Web 服务, 客户能够找出关于 WSDL 和它要调用的服务信息, 使用 ServiceFactory 类来创建服务。JAX-RPC ServiceFactory 获得 JAX-RPC 服务, 它与厂商无关, 能够编写可移植代码。根据 WSDL2Java 生成的客户端构件获得服务的一个

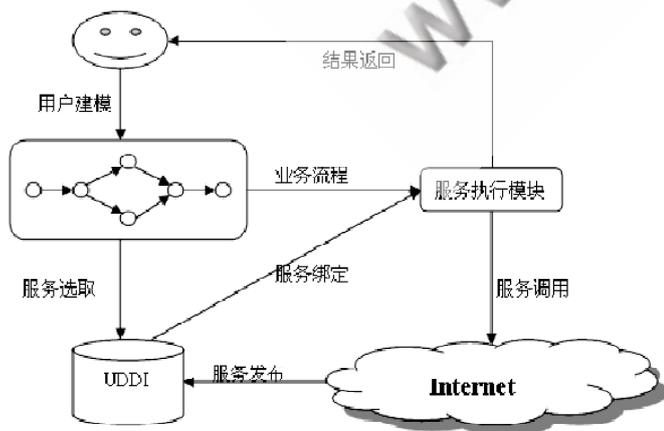


图 1 组合系统的执行流程

存根, 最后在该存根上调用 Web 服务总的操作; 最后服务执行模块将执行的结果返回给用户, 并用 XML 文件将其保存。

2 WSCS 中各个模块的设计

2.1 组合流程建模模块

组合流程建模模块为 Web 服务组合系统对外提供了图形化用户界面, 它可帮助用户完成整个业务流程的设计, 并动态生成服务选取模块所需的组合 Web 服务流程的描述文档。通过图形化建模与用户进行交互, 组合流程建模模块提供了一套简单并有效的操作方法, 使得用户可以根据自身的需求设计个性化的 Web 服务组合流程模型。

组合流程建模模块的运行流程: 首先用户利用图形化模型设计器设计流程, 设置流程执行的顺序; 其次流程执行顺序设计好后, 用户为流程中各图元, 尤其是节点配置相关属性, 设置合适的约束条件; 再次保存设计流程的组合文档和图形文档; 最后如果需要对流程进行修改, 则从文件夹中重新读取相关文档, 来生成可视化信息, 然后重复进行, 直到修改完毕并存档。

当满足用户需求的业务流程被设计好以后, 组合系统将会生成一个对这个业务流程进行详细描述 XML 文档, 以提交给 Web 服务动态选取模块和流程执行模块使用。组合 Web 服务模型文档描述了组合整个流程中各个节点在流程中的约束关系, 图形文档描述了流程中各图元的位置。

组合流程建模模块在整个系统中是一个重要的模块, 下面讨论模块的内部功能结构, 具体如图 2 所示。

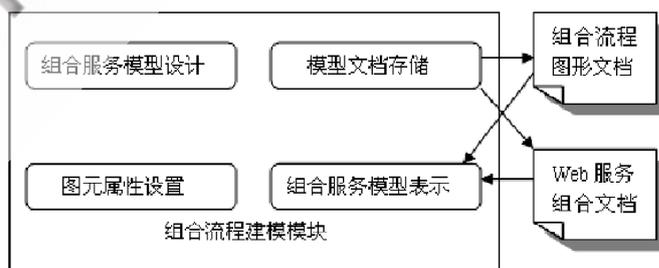


图 2 组合流程建模模块

(1) 组合服务模型设计中提供建模所需的各种图元(节点和弧)的绘制, 以实现业务流程模型的图形化界面, 设计者可以方便地对图元进行拖放、选取、删除等可视化操作。用户可以通过拖拉(Drag-Drop)方式定义新的流程或者修改已有的业务流程。

(2) 图元属性设置中, 系统对流程中各图元的属性进行设置。

(3) 模型文档存储、建模完成后, 要对生成的组合服务模型进行保存, 生成相应的文档。该文档以 XML 形式描述组合服务模型, 要与组合模型进行同步更新(如

用户对该文件的修改、读取等操作)。

(4)组合服务模型表示中包括文档解析,它可以从保存的组合流程图文档和 Web 服务组合文档中提取信息,再次把流程图形化地表示出来。

2.2 服务选取模块

服务选取模块的主要功能是根据设计的组合流程,动态地选取满足约束条件,并且服务质量最优的服务。即在流程投入运行前,根据流程设计时用户的需求,为每个活动节点选择合适的、可用的 Web 服务,生成可执行的 Web 服务序列。

Web 服务的描述是发现和使用该服务的基础,它需要能够被人和机器理解。服务描述文档中应包含功能性和非功能性需求(如响应时间,费用,可靠性,可用性等)信息,目前 Web 服务的描述采用 WSDL。

查找选取流程中所要用到的 Web 服务,可以从 UDDI 中进行查找,在 Web 服务标准体系中,UDDI 是一套基于 Web 的、分布式的、为 Web 服务提供的信息注册中心的实现标准规范,也可以从 Web 服务社区中查找。

Web 服务社区是根据服务功能分类的 UDDI,在服务选取模块中起着重要的作用,它把功能相同或相似的服务聚集在一起,这样服务选取就可以方便地到各服务社区中选取需要的服务,另外由于服务社区提供多个候选服务,服务选取就可以用上章讨论的算法,选择服务质量优的服务。系统到各服务社区中选取满足用户需求的服务,Web 服务发布在服务社区中,各个服务社区包含服务的描述文档和绑定地址。

Web 服务选取模块主要根据建模传过来的组合模型描述文档,分析其中的执行顺序和约束条件,然后采用 Bellman-Ford 算法依据 QoS 最优选出一组服务。服务选取模块主要结构如图 3 所示,流程如下:

(1)文档解析器解析建模系统传过来的组合服务模型文档。

(2)服务选取接收到解析后的文档按一定规则调用 UDDI 和服务社区中的查询接口返回的符合要求的服 务集,从全局优化的角度采用 Bellman-Ford 算法在服务社区中找到一组最优服务。

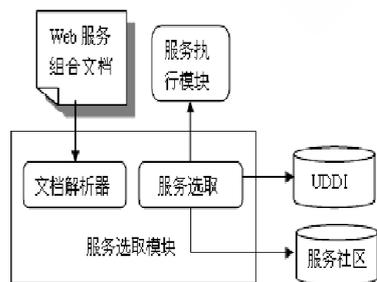


图 3 服务选取模块

(3)服务选取模块选取服务后,把节点与服务的地址相绑定,然后传给服务执行模块。

2.3 流程执行模块

流程执行模块把流程节点与具体的 Web 服务相绑定,然后按照流程进行服务调用,当组合 Web 服务的条件满足时,完成对服务的调度和执行,实现组合 Web 服务的功能。具体的运行工作主要是组合 Web 服务的调用,并维护其整个生命周期(包括创建、活动、出错、结束等)。

Web 服务组合流程在真正的投入运行之前,首先要进行服务的选取,为每个活动节点选择一个最佳的 Web 服务,这一功能由服务选取模块来实现。服务选取结束后,返回给流程执行模块一个用 XML 语言描述的组合流程文档。

执行流程的约束是对 Web 服务组合的整体限定,主要是对服务质量(QoS)的约束,对 QoS 的约束是指对 Web 服务组合整体性能的限定(如对 Web 服务组合的可用性、可靠性等性能的约束)。对 Web 服务组合 QoS 的约束直接影响了 Web 服务的选取。在选取 Web 服务时,既要满足对单个 Web 服务 QoS 的约束,也要考虑到对整个 Web 服务组合 QoS 的约束。

组合流程执行模块的主要功能是:解释流程定义、串行化流程、执行 Web 服务的调用,主要功能模块如图 4 所示:

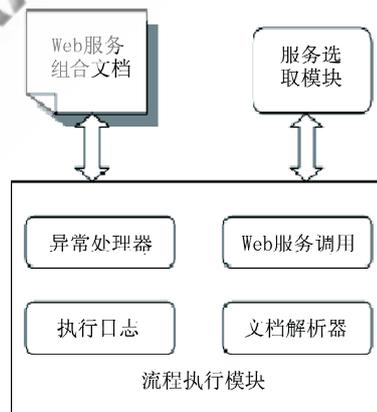


图 4 Web 服务组合系统流程执行模块

文档解析器:从组合流程中选出所要执行的流程文档,文档解析器首先分析该文档所描述的业务流程所代表的含义,各个服务的具体绑定地址,返回给服务调用并进而完成流程的实例化。

服务调用:一个流程要投入运行必须要有一个先后的顺序,在流程串行化时按照业务逻辑和一定的策略把一个流程串行化为 Web 服务调用序列,然后按照流程并行、串行、同步等待地调用各服务。

执行日志:存放执行过程中调用各个服务的具体信息,如果系统出错可以通过查看执行日志来帮助恢

复系统。

异常处理器：对于系统出现的故障，异常处理模块提供恢复处理手段。流程的执行可能会遇到各种各样的特殊情况发生(如网络超时、服务不可用)，而这些变化在流程设计时是无法预见的。

组合流程执行模块是整个运行系统的核心，它一方面为不同的 SOAP 引擎提供相应的调用接口；另一方面为不同的 Web 服务组合提供一个统一的运行环境；还为不同的 Web 服务的实现方式提供了统一的调用方式。

3 Web 服务组合系统的实现

目前，工程上科学计算的重要性显得越来越明显，大量的科学计算是普通机根本无法完成的，只有寻求大型机去解决。同时，大型机的用户也总会有时间给其他用户提供服务，以使大型机充分地利用。如何才能更好地发挥大型机的效率，使普通的用户也可以用大型机进行复杂的科学计算？可以把大型机上的科学计算封装成 Web 服务的形式，发布在 Web 上，这样用户可以方便地调用，既发挥了大型机的作用，又使普通机用户受益。

本文假设一个工程需要一个复杂的科学计算，它由下列一些科学计算组成：极限计算(limit calculus)、微分方程计算(differential equation calculus)、级数计算(series calculus)、矩阵计算(matrix calculus)。整个工程需要这些科学计算的组合，先进行极限计算，把结果传给下面微分方程计算和级数计算，然后再执行矩阵计算。把这 4 种计算封装成 Web 服务，现在的问题就是要对基本服务进行组合。

WSCS 先对组合服务进行建模，图形用户界面的建模工具对组合服务进行建模，建立基本服务的流程，特别的是一个同步节点，它要等待前面所有服务的结果都完成才进行下一个服务的执行，如图 5 所示。服务组合模型建好后，Web 服务组合系统自动把组合模型保存为 XML 文档，以便服务的调用执行。文档解析器就是用来保存和解析组合模型文档。

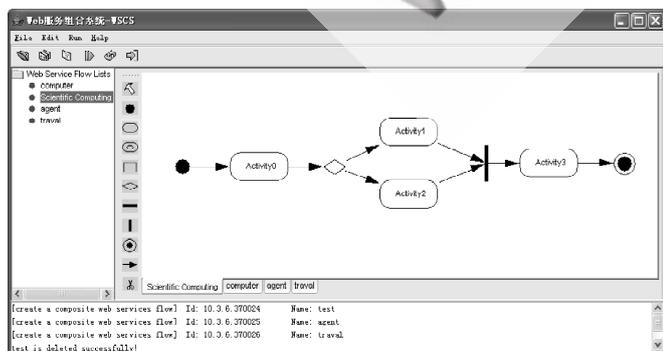


图 5 Web 服务组合系统的运行

在上面的文档中包含了需要的各个基本服务以及

各服务之间的执行流程。系统还提供了另外一个 XML 文档，它用来保存各个服务节点在图形用户界面中的位置，文档解析器会根据 XML 文档中信息，还原图形化流程，以便于以后服务流程的修改。

本文通过研究 Web 服务组合与工作流的异同点，借鉴工作流的参考模型及其体系结构设计了一个层次化的、统一的 Web 服务组合系统，为 Web 服务的集成和协作提供了有力的支持。同时对 Web 组合系统的各个子模块进行设计，最后给出了 Web 服务组合系统(WSCS)的原型，并且用了科学计算服务进行了 Web 服务组合。

Web 服务组合系统(WSCS)也存在一些不足之处，它首先没有为流程中的事务、安全等其他非功能特性提供支持，也没有支持灵活的异常处理机制，这在一定程度上限制了组合服务的能力；其次没有提供一个 Web 服务的性能评估和性能预测模型，尚不能根据单个服务节点的执行效率预测组合后的服务执行效率，这将是笔者下一步的研究工作。

参考文献

- [1] 岳昆, 王晓玲, 周傲英. Web 服务核心支撑技术: 研究综述[J]. 软件学报, 2004, 15(3): 428-442.
- [2] 李曼, 王大治, 杜小勇, 等. 基于领域本体的 Web 服务动态组合[J]. 计算机学报, 2005, 28(4): 644-650.
- [3] DAVIES N J, FENSEL D, RICHARDSON M. The Future of Web Services [J]. BT Technology Journal, 2004, 22(1): 118-130.
- [4] MILANOVIĆ N, MIROSLAW M. Current Solutions for Web Service Composition [J]. IEEE Internet Computing, 2004, 26(5): 51-59.
- [5] MEDHAHED B, BOUGUETTAYA A, ELMAGARMID A K. Composing Web Services on the Semantic Web [J]. The VLDB, 2003, 12(4): 333-351.
- [6] STEFAN T, RANIA K, THOMAS M. Composition of Coordinated Web Services [C]. IFIP International Federation for Information Processing, 2004, 294-310.
- [7] BOUALEM B, MARLON D, QUAN Z. Declarative Composition and Peer-to-Peer Provisioning of Dynamic Web Services [C]. In the Proceedings of the 18th International Conference on Data Engineering, 2002.
- [8] 张智, 李瑞轩. 基于对等网的 Web 服务发布和发现机制研究[J]. 计算机工程与设计, 2006, 27(16): 2949-2951.
- [9] MASSIMO M, ALESSANDRA M, ALESSANDRO P. Declarative Policies for Web Service Selection [C]. In the Proceedings of the Sixth IEEE International Workshop on Policies for Distributed Systems and Networks, 2005.
- [10] 谷清范, 吴介一, 张飒兵. 网格调度机制研究综述[J]. 计算机

(下转第 48 页)

ength	Frame control field					Sequence number	Source PAN	Source Address	Superframe specification				Beacon payload			Beacon Payload								
24	Type	Sec	Pnd	Ack.req	Intra.PAN		0x002A	0x0000	B0	S0	F.CAP	BLE	Coord	Assoc	00	21	84	2A	00	12	P.Id	Stk	Prof	P.Ver
	BCN	0	0	0	0	0x70	0x002A	0x0000	15	15	15	0	1	1	13	14	15	16	17	0x00	0x1	0x2		
ength	Frame control field					Sequence number	Dest. PAN	Dest. Address	Source PAN	Source Address	Association request													
21	Type	Sec	Pnd	Ack.req	Intra.PAN		0x002A	0x0000	0xFFFF	0x3030303030303031	Alt.coord	FFD	Power	Idle.RX	Sec	Alloc.addr								
	CMD	0	0	1	0	0xC4	0x002A	0x0000	0xFFFF	0x3030303030303031	0	0	0	0	0	1								
i)	Length	Frame control field					Sequence number	RSSI (dBm)	FCS															
26	5	Type	Sec	Pnd	Ack.req	Intra.PAN		0xC4	-74	OK														
		ACK	0	0	0	0	0xC4	-74	OK															
ength	Frame control field					Sequence number	Dest. PAN	Dest. Address	Source Address	Data request	RSSI (dBm)	FCS												
18	Type	Sec	Pnd	Ack.req	Intra.PAN		0x002A	0x0000	0x3030303030303031		-72	OK												
	CMD	0	0	1	1	0xC5	0x002A	0x0000	0x3030303030303031		-72	OK												
		ACK	0	1	0	0	0xC5	-74	OK															
ength	Frame control field					Sequence number	Dest. PAN	Dest. Address	Source Address	Short addr	Assoc. status	RSSI (dBm)	FCS											
27	Type	Sec	Pnd	Ack.req	Intra.PAN		0x002A	0x3030303030303031	0x171615141312002A	0x796F	Successful	-74	OK											
	CMD	0	0	1	1	0x17	0x002A	0x3030303030303031	0x171615141312002A	0x796F	Successful	-74	OK											

图6 协调器与终端设备关联过程

首先, Zigbee 协调器上电以后, 不断周期发送空的数据包, 在允许的通道内搜索其他的 Zigbee 协调器, 并基于每个允许通道中所检测到的通道能量及网络号, 选择唯一的 16 位 PAN ID, 建立自己的网络^[4]。一旦一个新网络被建立, Zigbee 路由器与终端设备就可以加入到网络中了。而终端设备上电以后, 重复发送信标请求, 要求加入到最近的网络中。当协调器发现终端设备发出的信标请求, 则响应一个超帧结构, 用于设备间的同步, 一旦同步成功, 则实现图 5 中的关联过程, 由终端设备向协调器发送关联请求, 协调器同意则回应终端设备并自动分配 16 位的短地址, 至此, 两者组网成功。

网络形成后, 可能会出现网络重叠及 PAN ID 冲突的现象。协调器可以初始化 PAN ID 冲突解决程序, 改变一个协调器的 PAN ID 与信道, 同时相应修改其所有的子设备。通常, Zigbee 设备会将网络中其他节点信息存储在一个非易失性的存储空间——邻居表中。加电后, 若子设备曾加入过网络, 则该设备会执行孤儿通知程序来锁定先前加入的网络。接收到孤儿通知的设备检查它的邻居表, 并确定设备是否是它的子, 若是, 设备会通知子设备它在网络中的位置, 否则子设备将作为一个新设备来加入网络。而后, 该子设备将产生一个潜在双亲表, 并尽量以合适的深度加入到现存的网络中^[5]。

通常, 设备检测通道能量所花费的时间与每个通

道可利用的网络可通过 ScanDuration 扫描持续参数来确定, 一般设备要花费 1 min 的时间来执行一个扫描请求, 对于 Zigbee 路由器与终端设备来说, 只需要执行一次扫描即可确定加入的网络。而协调器则需要扫描两次, 一次采样通道能量, 另一次则用于确定存在的网络。

限于篇幅, 本文没有列出各种帧结构以及消息的处理过程, 对于 Zigbee 协议的具体应用即是对数据包的封装与分解, 这些分析、学习对于编写上位机软件, 开发通过 Zigbee 协议与上位机进行交互的轻量级的现场设备, 都是有意义的。已经据此以 C++ 开发出用户自定义的上位机程序, 实现对传感器数据的监视和现场采集、数据库入库、趋势图等功能。

参考文献

- [1] ZigBee Alliance. ZigBee Document 053474r13 [S]. December 1, 2006
- [2] ZHENG JianLiang, Lee Myung. A Comprehensive Performance Study of IEEE 802.15.4[M]. IEEE Press Book, 2004.
- [3] Chipcon, Packet Sniffer for IEEE802. 15. 4 and Zigbee [S]. User Manual. Oslo, Norway, Oct. 2004.
- [4] KINNEY P. Zigbee Technology : Wireless Control that Simply Works [S]. Zigbee Alliance , Oct . 2004.
- [5] Zigbee Alliance. Network Specification (Draft Version 1.0)[S]. 2004.

(收稿日期: 2008-11-20)

(上接第 44 页)

应用研究, 2006, 23(5):1-4.

- [11] 官荷卿, 张文博, 魏俊, 等. 一种应用敏感的 Web 服务请求调度策略[J], 计算机学报, 2006, 29(7): 1189-1198
- [12] 单志广, 林闯, 肖人毅, 等. Web QoS 控制研究综述 [J]. 计算机学报, 2003, 27(2): 145-156.

- [13] ZENG LangZhao, BENATALLAH B, DUMAS M. Quality Driven Web Services Composition [J]. IEEE Transaction on Software Engineering, 2004, 30(5):311-327.

(收稿日期: 2008-12-03)