

DOI: 10.7652/xjtuxb201302001

一种敏捷服务组合方法模型的研究与设计

史旻¹, 邱劲锋¹, 侯迪¹, 齐勇¹, 林秦颖²

(1. 西安交通大学电子与信息工程学院, 710049, 西安; 2. 空军工程大学工程学院, 710051, 西安)

摘要: 针对目前服务组合技术复杂性高、成本高以及灵活性不足等问题, 提出一种敏捷服务组合方法模型。该模型以 Web 服务组件作为服务组合的基本单元, 摆脱了传统服务组合对专业技术人员的依赖; 通过设计一种轻量级流程控制器, 使得服务组合流程不依赖于任何重量级的服务器, 组合结果得以实时反馈; 通过 Web 服务组件部署器的设计, 实现了基本的服务组合流程可被更复杂的服务重用。通过在企业应用集成项目中的应用, 验证了该模型所实现的服务组合具有更好的易用性和灵活性。

关键词: 服务组合; Web 服务组件; 实时反馈

中图分类号: TP311 **文献标志码:** A **文章编号:** 0253-987X(2013)02-0001-06

Design of an Agile Service Composition Approach Model

SHI Yi¹, QIU Jinfeng¹, HOU Di¹, QI Yong¹, LIN Qinying²

(1. School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China;

2. College of Engineering, Air Force Engineering University, Xi'an 710051, China)

Abstract: An agile service composition approach model is proposed to focus on the problem that most current service composition techniques are of high complexity, high cost and less flexibility. The algorithm uses Web service components as the basic service composition units to shake off reliance of traditional techniques on professionals. A light flow controller is designed to make the service composition flow not rely on any heavy server and feedback in time. Basic service composition flow can be reused in more complex services by designing a Web service component deployer. The usability and flexibility of the proposed model are verified by implementing it in an enterprise integrated project.

Keywords: service composition; web service component; on-the-fly feedback

分布式计算研究领域的重要计算模式是面向服务的计算^[1-2], 即服务组合技术。依据应用领域的不同, 服务组合技术可以分为^[3]: 面向流程的服务组合, 如 BPEL4WS^[4]、WSCI^[5]、BPML^[6] 和 WS-CDL^[7]; 面向数据的服务组合, 如 AXML^[8]、WWT^[9]; 面向语义的服务组合, 如 DAML-S/OWL-S^[10]。

现有的常用服务组合模型能较有效地整合遗留应用, 生成反映业务逻辑的流程。但是, 这些模型在实际应用中存在局限性, 主要表现在以下几点。

(1) 服务组合复杂性高。现行的服务组合技术和服务组合方法都侧重于软件层面, 需要软件开发专业人员的参与, 应用系统的开发与运行难度大、周期长, 业务人员难以直接参与服务组合, 容易造成逻辑层难以满足业务层频繁变化的要求, 出现业务层与逻辑层不一致的情况。

(2) 服务组合灵活性不足。目前, 动态的服务组合虽然可以自动地选择、绑定 Web 服务, 自动地创建过程模型, 但这需要开发者事先指定一些约束关

系,一旦这些约束关系和用户的偏爱发生改变,则整个应用也要进行重新修改部署,造成业务的中断,因此这种动态性是有限的,应用的即时集成仍旧是一个挑战性问题。另外,这些服务组合方式比较适合在业务比较稳定、性能要求相对较高的大企业中实施,而对于业务需求经常变化的中小型企业,应用时就显得灵活性不足。

(3)服务组合成本高。现在常用的服务组合方式除了要求有很高的技术支持外,一般都要求有很高的软硬件支持,包括中间件平台、流程运行引擎等重量级服务器,如IBM的Websphere Server、Websphere Process Server等,相应的软件使用维护复杂,成本高。

由于上述局限性,导致这些服务组合模型目前没有在工程界,特别是在业务灵活多变的中小型企业中得到大规模的使用。

本文针对上述服务组合模型的局限性,结合Web2.0和传统的服务组合技术,提出一种敏捷服务组合方法模型。通过该模型,使得服务组合的主体不再是专业技术人员,普通的业务人员就可以自主进行服务组合,实现业务层和逻辑层的统一;服务组合运行不依赖于任何重量级的运行环境,组合结果变更可以实时反馈,具有更好的服务组合灵活性和应用级的重用性。以该模型系统为试验平台,通过在企业应用集成项目中的应用,验证了其可行性及有效性,并通过与传统服务组合方法的比较,证明该模型提供了一种业务端用户参与、简单易用的服务组合机制,具有更好的灵活性和适用性。

1 敏捷服务组合方法模型基本思路

传统的服务组合技术以技术人员为中心,在满足用户的即时需求方面还欠缺灵活性,而Web2.0技术^[11]的出现则为服务组合方法提供了新的思路。

首先,Web2.0贯穿“以用户为中心”的理念,充分强调用户的体验、参与和自主性。可以借鉴这种思想来实现以业务端用户为中心的服务组合,对原有的服务表现形式进行改造,提供一种业务端用户可理解、友好展示、易于与服务进行互操作的服务表现形式,使得用户更多地参与到服务组合中去,最快地实现从业务到逻辑的转化。Web2.0提供的Ajax^[12]、Dojo^[13]等数据异步交互技术,以及注重Web层内部数据联系等技术,都为这种改造提供了条件。

其次,Web2.0强调运行简单、业务逻辑更改反

应灵敏的特性,用户能够尽可能地自主更改逻辑,并要求快速得以反馈。当前的服务组合编排结果需要重量级的底层服务器的支持,而通过借鉴Web2.0的轻量便捷思想,可以设计轻量级的流程运行机制,并且使服务组合变更可以实时反馈,不用进行重新载入应用等操作,提高服务组合的柔性和业务应用的灵活性。

最后,Web2.0所强调的“人人为我,我为人人”的思想,本质上是强调资源的良好重用性,用户不仅是内容的消费者,同时也是内容的提供者。在更多强调用户参与的服务组合中,可以提供机制来保证新型的基于应用级服务内容的重用性。

敏捷服务组合方法模型主要解决以下关键问题。

(1)满足即时需求的灵活性。要满足用户的即时需求,就要对传统的服务组合结果运行机制进行改进,保证在服务组合结果变更时,能够进行快速反馈。为解决此问题,需要设计新型的流程运行控制方式,其特点如下:①分散的流程控制方式,对不同的流程进行独立控制,保证不同流程之间运行环境的互不干扰,一个流程的变更不会影响到其他流程的运行;②流程运行于轻量级运行机制,不依赖于任何重量级的工作流引擎;③流程运行时动态实时载入流程定义文件,实时反映服务组合变更结果,避免业务更改引起的应用重载,保证流程运行的连续性,满足用户的即时需求。

(2)易用性。要由业务终端人员而不是专业技术人员进行服务组合,缩短从业务层面到技术层面的映射,就需要考虑如何降低对用户的编程要求,包括:①服务显示图形化,通过对传统服务表现形式进行改造,提出了Web服务组件的概念,可对服务进行友好展示,便于用户的识别和使用;②采用应用级的服务组合,与传统接口级的服务组合不同,Web服务组件是应用级的,需要提供一套机制,使得Web服务组件之间能够进行组合,实现业务流程,用户不用像技术开发人员一样进行接口编程,就可以设定所有的数据逻辑和控制逻辑;③监控问题,实时察看服务的执行情况,对服务组合运行数据进行监控,及时获得异常信息并对应用流程进行调整。

(3)基于应用级的服务重用。与接口级的传统服务重用不同,在敏捷服务组合方法模型中,通过应用级的服务组件(Web服务组件)部署机制设计,实现服务组件的注册与发布,使得单元Web服务组件和组件组合结果都可以被重用,从而实现应用级的服务重用。

2 敏捷服务组合方法模型

2.1 模型框架

首先,通过页面适配器的设计,把屏幕抓取的内容转化为标准可用的 Web 服务组件,或者自主对 Web 服务组件进行编写,通过这 2 种方式向外提供可重用的 Web 服务组件,通过对已有的服务组件进行部署,生成组件描述文件,单元组件组合后的子应用也可再被重用。然后,由普通用户基于业务需求

对现有已部署的 Web 服务组件进行组合,组件服务组合底层支持机制自动生成代表一定功能逻辑的业务流程;流程运行底层支持控制流程实例的创建和运行,对流程描述文件进行动态载入,可实现服务组合变更实时反馈,是一种新型的分散式轻量级的控制方式。最后,流程监控器对流程运行状况和运行的结果数据进行监控,并以图形化的方式体现。敏捷服务组合方法模型框架如图 1 所示。

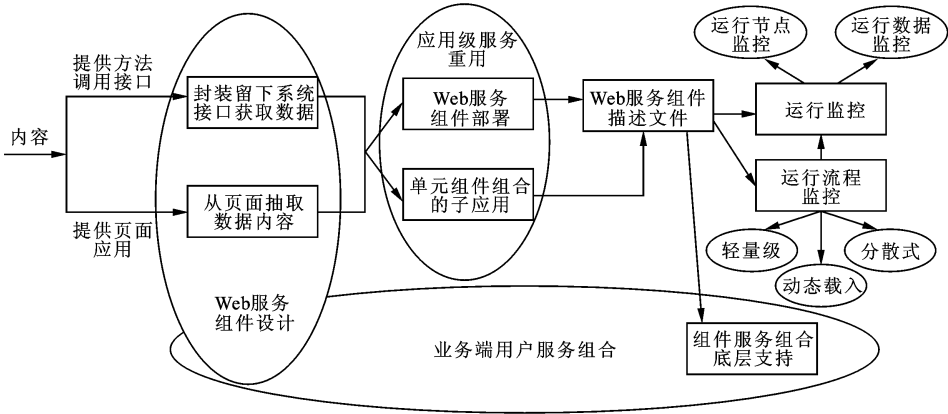


图 1 敏捷服务组合方法模型框架

2.2 业务端用户服务组合

业务端用户服务组合是指由具备较少或者不具备 IT 专业知识的业务用户在不需要 IT 专业人员帮助的前提下来自主进行服务组合。要使业务端用户能够自主进行服务组合,一方面要对传统服务的表现形式进行改造,通过 Web 服务组件的设计,不仅包含服务的特性,同时提供友好可视化的服务表现形式,使得业务端用户可以对服务进行识别,并通过在页面中拖拽 Web 服务组件进行服务组合;另一方面,要设计一套底层机制,使得 Web 服务组件之间能够进行组合,以实现业务流程。

2.2.1 Web 服务组件 Web 服务组件为业务端用户提供了底层服务资源的可视化表示形式,这种表示能够体现业务领域的内容,被业务用户所理解与使用。Web 服务组件的结构设计如图 2 所示。

每个 Web 服务组件的结构设计包括如下 3 层。
(1)表示层。Web 服务组件是对传统 Web 服务的改造,使服务“可显示化”,使得用户可以直接获取并使用服务的内容,与服务进行互操作。Web 服务组件是运行在页面中的简单 Web 应用,表示层负责 Web 服务组件在页面中的显示。

(2)数据层。数据层提供了组件的数据内容,为

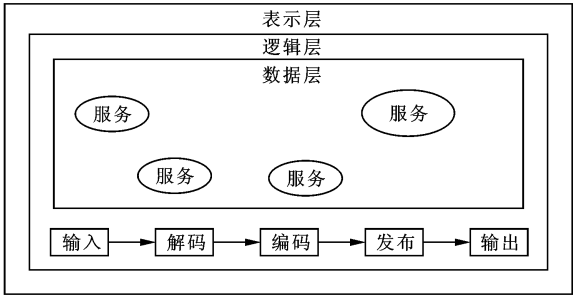


图 2 Web 服务组件的结构

服务内容的揉合提供了基础。数据层一般由所提供的常规服务调用接口组成,一个 Web 服务组件可包含多个常规服务,对于一些遗留应用,没有提供可调用的服务接口,但提供了基于 B/S 结构的 Web 应用,此时,数据层可以通过抓取 Web 页面屏幕内容的方式予以实现,同时每个 Web 服务组件代表一定的业务意义,从这个角度可以认为 Web 服务组件提供的是粗粒度业务级的服务。

(3)逻辑层。逻辑层包含 Web 服务组件内部的数据逻辑,以及为保持不同组件之间数据传递格式的一致性,对组件输入输出数据所做的一些预处理。不同 Web 服务组件的数据都以同一标准的 JSON 格式来描述。

由以上叙述可以看出,敏捷服务组合方法模型的基本元素是对传统 Web 服务进行了改造的粗粒度应用级的 Web 服务组件,Web 服务组件对传统的服务进行“业务化”,为业务端用户能够自主进行服务组合提供了基础。

2.2.2 基于 Web 服务组件的应用级服务组合 在敏捷服务组合方法模型中,流程编排的基本元素是 Web 服务组件,要设计基于这些应用级 Web 服务组件的服务组合底层机制,支持应用级元素的流程编排。

服务组合是通过定义 Web 服务组件之间的关系实现的。根据上面定义的 Web 服务组件,每一个组件包含有输入接口和输出接口,如图 3 所示。

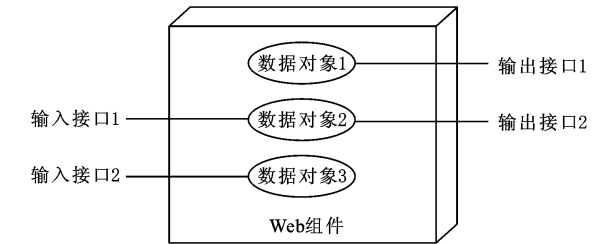


图 3 Web 服务组件对外接口

如果一个组件的输出对应另一个组件的输入,就定义了 2 个组件之间的先后关系。每个输出可以对应多个输入,每个输入也可以对应多个输出,由此构成了流程多条可能的路径。这是一种数据驱动的服务组合,运行时数据的变化决定了流程的运行路径。同时,还要考虑到数据对象的映射问题。服务组合的结果生成关于该流程的描述文件。

2.3 分散式流程运行控制

在敏捷服务组合方法模型设计中,为流程的运行与管理设计了流程控制器。流程控制器在形式上是一个普通的处理流程运行逻辑的 PHP 类文件,和所有普通的 PHP 类文件一样,它运行在简单的 APACHE 服务器上,无需其他任何流程引擎平台服务器支持。当用户完成了服务组合、流程编排后,将自动生成一个流程描述文件和一个流程控制文件,每一个流程控制文件控制相应流程的运行,流程和流程控制器是一一对应的关系。与传统的流程引擎集中控制方式有很大的不同,这是一种分散的流程控制方式,每个流程对应一个流程控制器,不同流程的运行互不影响。传统流程引擎与流程控制器的区别如图 4 所示。

2.4 组合变更实时反馈

当业务需求发生改变时,需要对服务组合结果

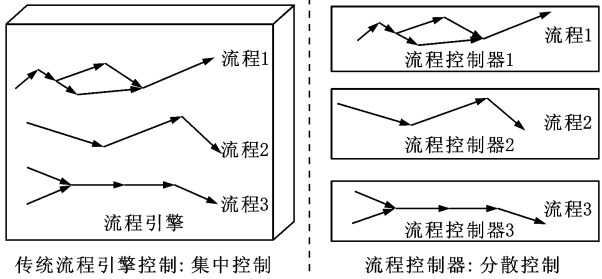


图 4 传统流程引擎与流程控制器的区别

进行修改。传统的服务组合方式一般需要经过重启运行服务器、重新部署应用、启动应用等步骤,而敏捷服务组合方法模型设计避免了这些步骤,服务组合的修改可以实时反馈,从而可增加服务组合的灵活性,降低动态变化的成本,满足用户的即时需求。

服务组合变更的实时反馈^[14]是通过以下 2 个方面来实现的。一方面,是流程控制方式的改变。从上面的分析可以知道,流程的运行是通过流程控制器来实现的,这是一种分散的流程控制,多个流程之间不依赖于任何共同的重量级流程平台。传统的流程一般是一个 J2EE 的应用,需要部署到服务器中,服务器为该应用的运行提供环境,而流程控制器是用解释性语言 PHP 实现的,采取解释性的方式控制流程,不存在应用重新部署的问题,保证了应用的连续性。另一方面,流程控制器根据流程的描述文件决定流程的运行情况,服务组合变更的结果体现在流程描述文件的变化上;流程控制器对流程文件采取实时加载的策略,每当要决定下一个节点进行节点转换时,实时载入流程描述文件并据此作出决定。通过以上 2 个方面,保证了服务组合变更的实时反馈。流程描述文件实时载入机制的原理如图 5 所示:当服务组合变更时,流程描述文件自动进行修改;当流程运行时,该流程的控制器动态绑定该描述文件,实时查询下一个运行节点。

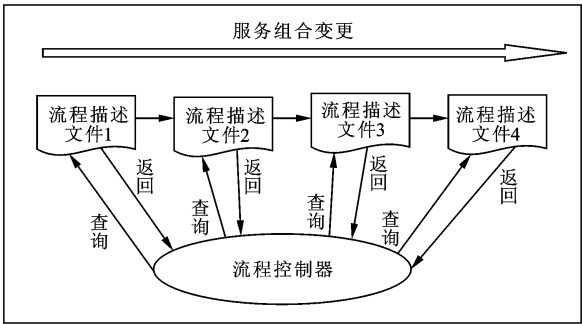


图 5 流程描述文件实时载入机制的原理

2.5 基于应用级的服务重用

在传统服务组合中,通过 UDDI 等服务注册机制,实现了服务的透明访问与重用。在敏捷服务组合方法模型中,通过 Web 服务组件部署机制设计,Web 服务组件以及由这些组件组合后代表一定业务流程的应用都可以被重用,这与接口级的传统服务重用不同,是一种应用级的服务重用。

首先,通过灵活的 Web 服务组件部署机制设计,用户可以上载 Web 服务组件,Web 服务组件的部署可以看成是服务的注册与发布,其运行是无状态的,每一次运行都进行实例化,这就使得同一组件可以被不同的应用所重用。其次,单元 Web 服务组件经过服务组合后所形成的流程应用,在对外的数据接口表示上与一般的 Web 服务组件是一致的,可以看作是一种功能更为复杂、粒度更粗的特殊 Web 服务组件,作为应用子流程被重用。

2.6 运行监控支持

流程监控器对流程运行的各个方面进行图形化的监控,观察流程运行的正确性、性能与瓶颈,为流程的维护与改进提供依据。流程监控主要包括流程运行节点监控和流程运行数据监控。流程运行节点监控实时反映流程中节点运行的先后变化状况。流程运行数据监控主要包括以下 4 方面数据的监控。

(1)节点平均运行时间监控:以图形化的方式实时显示同一种流程实例所经过的每个节点(所有该流程实例可能经过的节点)被调用运行的次数、运行总时间和该节点的平均运行时间。

(2)流程实例退出节点监控:对最后的退出节点进行统计,以反映流程的运行情况。根据不同的情况,流程实例可能在不同的节点结束运行,而最后一个节点往往能反映该流程实例运行的正确性。

(3)节点吞吐量监控:节点吞吐量反映节点随时间被调用的变化情况,通过以图形化方式显示所有节点的吞吐量,可以形象地观察节点的运行情况,分析出关键组件和瓶颈节点。

(4)流程吞吐量监控:以图形化的形式形象地显示在某个时间区间内正在运行的和运行结束的流程实例数,及其随时间的变化情况。流程实例分为正在运行和运行结束 2 种状态,流程吞吐量是随时间变化的流程实例状态统计。

通过以上监控,可以形象地观察流程运行的正确性、性能与瓶颈,为流程的维护和改进提供依据。

3 模型应用及结果分析

将敏捷服务组合方法模型应用于某公司的应用

集成项目——ERP(Enterprise Resource Planning)系统与 CRM(Customer Relationship Management)系统的应用集成,并且验证其可行性和有效性。

3.1 应用实例

某公司先后引进并实施了一种 ERP 系统和一种 CRM 系统(On Demand CRM)。销售人员在任何时间和地点,只需要通过普通的 Web 浏览器就可以使用和管理客户及销售信息,包括客户信息、商机、业务机会,以及客户及销售信息分析图表等。CRM 系统中维护的业务机会和 ERP 系统中维护的销售订单有着非常紧密的关系,公司希望销售人员能够把业务机会和销售订单有效地整合起来,也就是希望能够将这 2 套系统整合应用,从而进一步提高业务运作的效率。

3.2 解决方案

利用敏捷服务组合方法模型可以较好地解决以上问题。具体解决方案的环境配置为:整个系统运行于 3 台微机上,构成一个集成环境;3 台微机的配置均为 PIV 1.6 GHz CPU,512 MB SDRAM,分别安装 ERP、CRM 应用系统和这 2 个系统的集成应用。所有的计算机均通过 100 Mb/s 以太网、使用神州数码 DCS-1016 10/100 Mb/s 小型交换机连接。操作系统为 RedHat Linux AS 4,Web 服务器为 Apache 2.2.0,语言环境为 PHP 5.1.2,数据库使用 Mysql5.0。与传统的工作流服务组合实现方式相比,敏捷服务组合方法模型大幅度减少了应用整合的软硬件成本。

3.3 结果分析

通过运行敏捷服务组合方法模型,获取的数据如图 6 所示。由图 6 可以看出,在流程中:创建业务机会节点的总运行时间和平均运行时间最长,这是因为销售人员需要在该节点进行长时间的人机交互,用以填写业务机会的基本信息、显示库存及显示用户详细信息等,可以考虑进一步对该节点进行细分,分解为多个子节点;维护业务机会节点的调用次数最多,因为该节点在一次流程实例中可能与业务人员进行多次交互。节点平均运行时间的监控结果符合业务实际情况,验证了流程运行的正确性。此外,流程实例运行成功创建了订单的占到了 80%左右,也符合实际情况。

下面分析敏捷服务组合方法模型是否解决了第 1 节中提出的拟解决的 3 个关键问题,这有助于评价该模型是否满足了设计目标。

首先,通过轻量级流程控制器的设计,使得服务

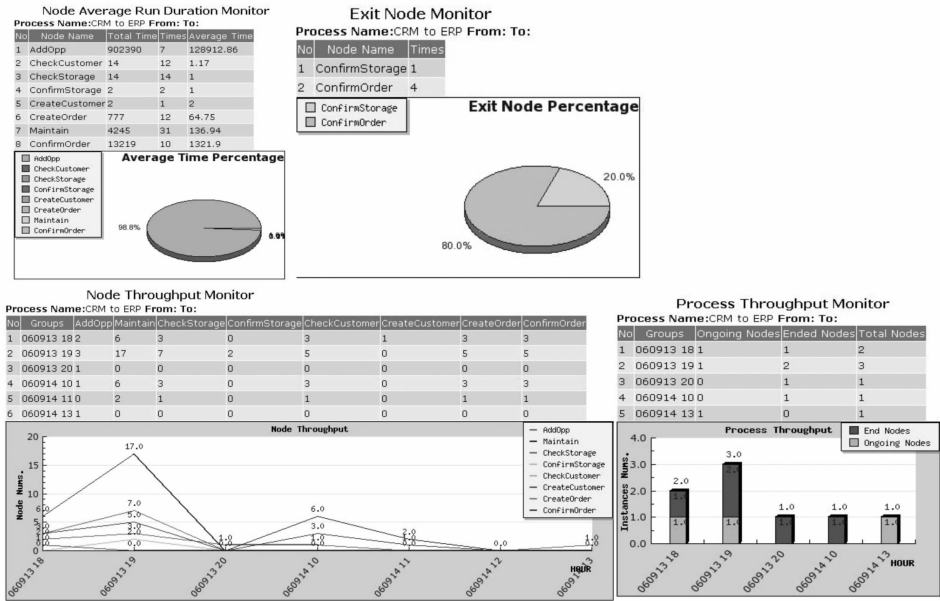


图 6 模型应用结果数据分析

组合流程不依赖于任何重量级的服务器。例如在本例中,如果不进行“库存检查”,则用户只要不定义“库存检查”节点的先后关系即可,组合结果可以实时反馈。所以,敏捷服务组合方法模型解决了满足用户即时需求的问题。

其次,通过对传统服务进行改造,将 Web 服务组件作为服务组合的基本单元,摆脱了传统服务组合对专业技术人员的依赖,解决了敏捷服务组合方法模型易用性的问题。在本实例中,公司的业务人员或者咨询人员可以通过定义 Web 服务组件间的关系来定义业务,并且可以容易地对业务进行更改。

最后,通过 Web 服务组件部署器的设计,使得 Web 服务组件和 Web 服务组件组合的结果都可以被重用,基本的服务组合流程可以被重用到更为复杂的服务中去。

通过模型应用实例的实施运行和对运行结果数据的监控分析,我们认为:敏捷服务组合方法模型解决了 CRM 系统和 ERP 系统的整合问题,实际应用效果良好,销售人员能够自主进行服务整合,并能够灵活地对业务流程进行更改,从而验证了敏捷服务组合方法模型的可行性与有效性,同时,这些分析数据为业务人员进一步优化流程提供了依据。

4 结 论

敏捷服务组合方法模型的适用对象是业务人员,他们在进行应用构建时不需要了解每个 Web 服务编程接口的具体信息,可以通过在页面中拖拽

Web 服务组件来定义不同组件之间的关系,完成服务组合的定义。这种方式简化了用户定义流程的复杂度,避免了逻辑实现层与业务层的偏差。敏捷服务组合方法模型开发平台提供了服务变更实时反馈机制,支持运行时构造应用,实现了服务组合的灵活性,可满足用户的即时需求。敏捷服务组合方法模型能够快速、低成本地构造应用,使敏捷服务组合不依赖于任何重量级的服务器,从而保证了服务组合的轻量级特性。

参考文献:

[1] 韩燕波,王洪翠,王建武,等. 一种支持最终用户探索式组合服务的方法 [J]. 计算机研究与发展, 2006, 43 (11): 1895-1903.
HAN Yanbo, WANG Hongcui, WANG Jianwu, et al. An end-user-oriented approach to exploratory service composition [J]. Journal of Computer Research and Development, 2006, 43(11): 1895-1903.
[2] PAPAZOGLU M P, GEORGAKOPOULOS D. Service-oriented computing [J]. Communications of the ACM, 2003, 46(10): 24-28.
[3] 任志宏. Web 服务复合的若干关键问题研究 [D]. 北京: 中国科学院研究生院, 2003.
[4] ANDREWS T, CURBERA F, DHOLAKIA H, et al. Business process execution language for Web services (BPEL4WS) version 1. 1 [EB/OL]. [2012-05-04]. <http://www-106.ibm.com/developerworks/web-services/library/ws-bpel/>.

(下转第 68 页)