

支持事务机制的 Web 服务组合 QoS 属性预测方法

吴江霞 杨放春

(北京邮电大学网络与交换技术国家重点实验室 北京 100876)

摘要: 该文提出了一种支持事务机制的 Web 服务组合 QoS 属性预测方法。分析了事务异常处理策略及其对组合服务执行流程的影响, 定义了支持事务机制的组合服务描述模型, 并基于此模型提出了组合服务 QoS 属性预测算法。实验证明该文方法在对具有事务机制的组合服务进行预测时, 准确度优于已有的工作流预测方法, 具有较好的可行性和有效性。

关键词: Web 服务组合; QoS 预测; 事务机制

中图分类号: TP393.07

文献标识码: A

文章编号: 1009-5896(2008)03-0703-04

QoS Prediction of Web Service Composition with Transaction Mechanism

Wu Jiang-xia Yang Fang-chun

(State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: An approach for QoS prediction of Web service composition with transaction mechanism is proposed. With the analysis on the exception handler policy of the transaction and the effect on the execution processes of composite services, the specification model of composite services is defined. On the base of the model, an algorithm to predict the QoS of composite services is proposed. The experiment shows the approach gets lower error rate than the existing workflow based prediction approach while estimating the composite service with transaction mechanism, and proves the feasibility of the algorithm.

Key words: Web service composition; QoS prediction; Transaction mechanism

1 引言

Web 服务组合往往从功能角度设计组合服务。对组合服务的执行期 QoS 属性进行预测, 可以判断该组合服务是否满足非功能需求, 或者从功能相似的组合服务中选择 QoS 较优的方案。目前的 Web 服务组合 QoS 预测方法有聚合函数预测方法^[1], 基于软件体系结构预测方法^[2], 模拟预测方法^[3], 工作流预测方法^[4-7]。其中工作流预测方法的可行性和准确度优于其它方法。

但以上方法都没有考虑事务处理机制对组合服务 QoS 的影响, 在对具有事务机制的组合服务进行预测时, 准确度十分有限。文献[8]中提出了失败感知预测方法, 分析了失败恢复对 QoS 的影响, 但没有考虑事务机制。而且由于输入参数很难获得, 其方法在实际中并不可行。

本文通过分析 Web 服务事务机制对组合服务执行流程的影响, 给出了根据事务异常处理策略列举执行流程的方法, 定义了支持事务机制的组合服务描述模型, 并基于此模型提出了组合服务执行期 QoS 属性均值预测算法。通过实验

证明了该方法的预测准确度优于现有未考虑事务机制的预测方法, 具有较好的可行性和有效性。

2 支持事务机制的组合服务描述模型

Web 服务组合的事务机制用于保证成员服务运行结果的一致性和可靠性。Web 服务事务是遵循扩展事务模型 ATM(advanced Transaction Model)的一组服务调用操作集合^[9], 满足 SACReD 属性, 即语义原子性(semantic atomicity), 一致性(consistency), 可恢复性(resiliency)和持续性(durability)^[10]。当事务执行发生异常时, 正常流程被中断, 根据预定义的异常处理策略, 进行补偿恢复。包括前向和后向两类补偿操作^[11]。前者通过重试或替换恢复发生了异常的操作。后者通过回滚撤销事务中已完成的操作。在对具有事务机制的组合服务进行评估时, 需同时考虑正常流程和异常处理流程。

下面是一个具有事务机制的服务组合实例。其功能是购买一套正装, 包括西服和领带。包括 4 项成员服务: 订单处理(orderreceipt), 西服订购(suitorder), 领带订购(tieorder), 和订购结果报告(orderresponse)。流程如图 1 所示。由于单独购买西服或领带没有意义, 二者要么都订购,

2006-09-22 收到, 2007-03-19 改回

国家 973 规划(2003CB314806)和教育部长江学者和创新团队发展计划资助课题

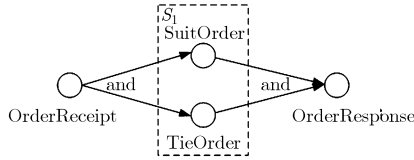


图1 正装购买组合服务流程

要么都不订购。因此 SuitOrder 和 TieOrder 构成一个事务。其异常处理策略为：当 SuitOrder 失败时，采用前向补偿，调用另一代理商的西服订购服务 SuitOrderAlter。当 TieOrder 失败时，采用后向补偿，调用 SuitOrderCancel 操作，撤销 SuitOrder 的操作。假定 SuitOrder 和 TieOrder 不会同时失败，而且异常处理操作不会失败。

定义如下组合服务描述模型。

定义1 组合服务流程 组合服务流程描述正常情况下成员服务调用顺序关系。采用工作流的形式描述^[5]，工作流任务节点表示服务调用，节点间的转移关系表示调用顺序关系。正装购买组合服务流程如图1所示。

定义2 范围 S 范围代表一个 Web 服务事务，包含一组任务节点，可以是组合服务流程的一部分或全部。一个组合服务可包含多个范围。范围之间可以任意深度嵌套，遵循事务嵌套规则^[12]。当范围执行发生异常时，根据预定义异常处理策略，进行补偿恢复。正装购买组合服务中，SuitOrder 和 TieOrder 构成范围 S_1 。如图1所示。

定义3 异常处理策略 EHP 表示范围的预定义异常处理策略集合。范围 S_α 的异常处理策略，记为 $EHP_\alpha = \{ehp_{\alpha 1}, ehp_{\alpha 2}, \dots, ehp_{\alpha n}\}$ 。ehp_{ai} 表示一条异常处理策略，由触发事件和处理行为组成^[11]，记为 $ehp_{\alpha i} = (event_{\alpha i}, action_{\alpha i})$ 。其中， $event_{\alpha i}$ 表示触发该策略的异常事件。可以为简单异常，即单个成员服务调用失败；或复合异常，即用“与”连接的简单异常。 $action_{\alpha i}$ 定义了当 $event_{\alpha i}$ 为真时的异常处理操作及其顺序关系。 $\forall event_{\alpha i}, ehp_{\alpha j} \in EHP_\alpha, i \neq j$, 有 $event_{\alpha i} \neq event_{\alpha j}$ 。

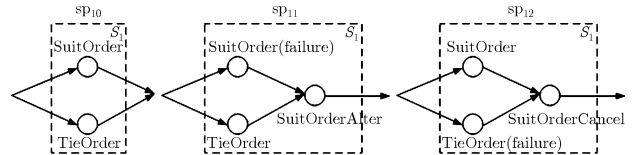
正装购买组合服务中，范围 S_1 的异常处理策略记为 $EHP_1 = \{ehp_{11}, ehp_{12}\}$ ，其中 $ehp_{11} = (Failure(SuitOrder), Alteration(SuitOrderAlter))$ ， $ehp_{12} = (Failure(TieOrder), Compensation(SuitOrderCancel))$ 。

定义4 范围流程 SP 设范围 S_α 的所有可能执行流程集合为 SP_α 。执行流程包括正常和异常两类。正常流程是指没有发生异常时范围的执行流程，记为 $SP_{\alpha 0}$ 。异常流程是指发生异常时范围的执行流程，包括异常处理前(PRE-EH)，异常处理(EH)和异常处理后(POST-EH)3部分。不同的异常事件及处理行为将导致不同的异常流程。

正常流程可根据组合服务流程给出。异常流程可根据异

常处理策略列出。设范围 S_α 的异常处理策略 $EHP_\alpha = \{ehp_{\alpha 1}, ehp_{\alpha 2}, \dots, ehp_{\alpha n}\}$ ，其中 $ehp_{\alpha i} = (event_{\alpha i}, action_{\alpha i})$ 。根据 EHP_α 的 n 条策略可列出范围的 n 种异常流程，分别记为 $sp_{\alpha 1}, sp_{\alpha 2}, \dots, sp_{\alpha n}$ 。 $sp_{\alpha i}$ 的 PRE-EH 部分根据 $event_{\alpha i}$ 列出：正常流程中， $event_{\alpha i}$ 中所有简单异常发生之前的流程，包括发生了异常的服务调用，以及异常发生时处于完成或激活状态的服务调用。EH 部分根据 $action_{\alpha i}$ 列出：异常处理操作表示为任务节点，操作间的顺序关系表示为节点间的转移关系。POST-EH 部分根据 $action_{\alpha i}$ 中最后执行的异常处理操作类型决定。采用前向补偿时，POST-EH 为正常流程中断位置之后的流程。采用后向补偿时，POST-EH 部分为空，即范围的后续流程被取消了。

据此可列出正装购买组合服务中，范围 S_1 的执行流程 $SP_1 = \{sp_{10}, sp_{11}, sp_{12}\}$ ，如图2所示。

图2 范围 S_1 的执行流程

3 支持事务机制的服务组合 QoS 预测算法

基于第2节中的组合服务模型，定义如下算法，用于预测组合服务某类 QoS 属性的均值。组合服务 QoS 包括多种类型，文献[1]中给出了5种类型，包括代价、时间性能、可靠性、可用性和声誉。算法独立于具体的类型定义，适用于各类 QoS 属性预测。

算法流程为：(1)计算组合服务流程中各范围的 QoS 值；(2)将范围转换为具有等价 QoS 值的任务节点；(3)计算转换后所得组合服务流程的 QoS 值。

范围 QoS 值通过计算可能执行流程的 QoS 概率平均值获得。设范围 S_α 的执行流程 $SP_\alpha = \{sp_{\alpha 0}, sp_{\alpha 1}, \dots, sp_{\alpha n}\}$ ， $q_{\alpha i}$ 和 $p_{\alpha i}$ ($i = 0, 1, \dots, n$) 分别表示 $sp_{\alpha i}$ 的 QoS 值和 $sp_{\alpha i}$ 发生的概率，满足 $\sum_{i=0}^n p_{\alpha i} = 1$ 。可得范围 S_α 的 QoS 值 SQ_α 为

$$SQ_\alpha = \sum_{i=0}^n (p_{\alpha i} q_{\alpha i}) \quad (1)$$

$q_{\alpha i}$ 采用工作流 QoS 预测方法获得^[5, 6]。这里对工作流 QoS 预测方法进行简单的介绍。工作流预测方法用于对采用工作流形式表示的 Web 组合服务流程进行 QoS 属性计算。方法将工作流的基本结构抽象为7种类型，如顺序，循环，并行，选择等，并给出了7种类型下各类 QoS 属性的聚合公式，即通过基本结构中的任务节点 QoS 属性值计算结构的

QoS 属性值。例如 $\sum x_i$ 和 $\sum p_i x_i$ 分别是性能属性在顺序和选择结构下的聚合公式,其中 x_i 代表节点的性能属性值, p_i 代表节点所在选择分支被执行的概率。工作流由基本结构嵌套构成,通过递归的聚合过程能够获得工作流的 QoS 属性值。采用工作流预测方法能够计算范围可能执行流程的 QoS 属性值。工作流预测方法需要获得任务节点的 QoS 属性值。文献[5]中给出了正常服务调用节点 QoS 值的获取方法。对于表示异常调用的节点,由于无法确定其状态,因此仍采用正常调用时的 QoS 值。对于表示异常处理操作的节点,其本质是调用具有前向恢复或后向恢复功能的 Web 服务或操作,可利用文献[5]中方法获取 QoS 值。

$p_{\alpha i}$ 计算公式如式(2)所示。

$$p_{\alpha i} = \begin{cases} \text{event}_{\alpha i} \text{发生的概率}, & i \neq 0 \\ 1 - \sum_{i=1}^n p_{\alpha i}, & i = 0 \end{cases} \quad (2)$$

$\text{event}_{\alpha i}$ 为简单异常时,其发生概率为相应服务调用任务节点的失败概率。 $\text{event}_{\alpha i}$ 为复合异常时,其发生概率的计算遵循 $P(A \cdot B) = PA \cdot PB$ 的法则。服务失败概率,记为 p ,可通过 3 种途径获得。一是服务提供者在服务接口中给出的失败概率,记为 p' ,二是统计服务执行历史数据得到的失败概率,记为 p'' ,三是统计服务在该组合服务中执行的历史数据得到的失败概率,记为 p''' 。有 $p = c_1 p' + c_2 p'' + c_3 p'''$ 。其中 $c_1 + c_2 + c_3 = 1$; 权值 c_i 反映了该项数据与失败概率估计值的相似程度。相似度越高,权值越大。一般情况下, p''' 项的相似度最高。某项数据不可获得时,可设其权值为 0。

对于嵌套的范围,采用递归的方法,从内向外逐层计算范围的 QoS 值。直至获得最外层范围的 QoS 值。

范围的转换是将组合服务中的非子范围转换为具有相同 QoS 值的节点。所有以该范围中节点为终端,以该范围之外节点为始端的转移,变换为以该任务节点为终端。而以该范围中节点为始端,以该范围之外节点为终端的转移,变换为以该任务节点为始端。最后采用工作流 QoS 计算方法计算转换后所得流程的 QoS,即为预测结果。

该算法在对无事务机制组合服务进行预测时,过程与工作流预测方法相同;在对具有事务机制组合服务进行预测时,增加的复杂度为 $O(n \cdot r) + O(r)$,其中 r 表示组合服务中的范围数目, n 表示所有范围的|EHP|均值。

以正装购买组合服务为例,预测其时间性能属性均值。时间性能属性定义为服务的执行时间,即从发出请求消息到收到响应消息之间的时间间隔^[1]。

利用成员 Web 服务执行历史数据,可得各任务节点时

间性能属性均值及失败概率,如表 1 所示。

表 1 任务节点时间性能及失败概率估计值

	Order Receipt	Suit Order	Tie Order	Order Response	SuitOrder Alter	SuitOrder Cancel
时间性能 (s)	3.9	7.7	5.4	2.8	9.1	9.7
失败概率 (%)	-	21.2	15.1	-	-	-

由表 1 中数据计算 $SP_1 = \{sp_{10}, sp_{11}, sp_{12}\}$ 中各流程的性能值及发生概率,如表 2 所示。利用式(1)可得范围 S_1 的时间性能值为 11.1。

表 2 范围 S_1 各执行流程时间性能及发生概率估计值

	sp_{10}	sp_{11}	sp_{12}
时间性能 (s)	7.7	16.8	17.4
发生概率 (%)	63.7	21.2	15.1

图 3 是范围 S_1 转换为等价节点后的组合服务流程。利用工作流 QoS 计算方法,可得该组合服务时间性能均值预测结果为 17.8。

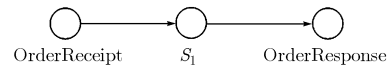


图 3 转换后的正装购买组合服务流程

4 实验及分析

为了验证预测算法的有效性和准确度,本文采用 JOpera 实现了正装购买组合服务。JOpera^[13, 14]为 Web 服务组合提供了图形化设计界面和执行平台。选用 JOpera 的原因是: JOpera 支持异常处理机制的实现,提供执行期间成员服务以及组合服务的时间性能数据记录,具有较好的可用性和稳定性。

本文记录了正装购买服务 15 次执行的时间性能数据,如表 3 所示。可以看到服务调用任务节点时间性能及失败率的估计值与实际值存在误差。其中 Failure (SuitOrder) 共发生 4 次,分别第 4, 6, 10, 15 次执行时,概率为 26.7%, Failure (TieOrder) 共发生两次,分别第 9, 11 次执行时,概率为 13.3%。

表 4 给出了组合服务时间性能实际值,本文方法及工作流预测方法的预测值,以及相应误差率。误差率计算公式为 $\eta = |V_e - V_t| / V_t$, V_e 和 V_t 分别表示预测值和实际值。

表3 正装购买组合服务执行数据

Instance	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Avg
Process	15.195	16.507	16.536	26.673	15.917	31.49	12.331	18.785	25.242	20.641	25.309	17.315	13.263	14.955	28.779	19.9
OrderReceipt	5.538	4.466	2.301	3.315	5.478	3.294	2.674	3.416	5.478	3.417	3.717	6.231	4.598	5.768	4.478	4.3
SuitOrder	7.324	9.868	9.062	3.322	6.424	13.002	7.524	9.266	5.264	12.932	7.124	8.86	6.335	6.964	11.96	8.3
TieOrder	6.166	6.606	3.562	4.024	8.165	4.124	5.524	7.246	11.062	5.566	10.365	4.242	6.531	5.326	5.164	6.2
OrderResponse	2.333	2.173	5.173	2.273	2.274	5.221	2.133	6.103	2.213	3.375	4.213	2.224	2.134	2.223	2.303	3.1
SuitOrderAlter	-	--	-	8.153	-	8.283	-	-	-	9.973	-	-	-	-	10.038	9.1
SuitOrderCancel	-	-	-	-	-	-	-	-	6.489	-	7.014	-	-	-	-	6.8

表4 预测结果比较

	实际值	本文方法预测值	workflow预测方法预测值
时间性能(s)	19.9	17.8	14.4
误差率 η (%)	-	10.6	28.2

可以看出本文方法的误差率明显低于 workflow 预测方法。在对具有事务机制的组合服务进行预测时，本文方法准确度优于现有预测方法。实验证明了本文方法的可行性和有效性。

5 结束语

通过分析 Web 服务组合中事务异常处理策略对组合服务执行流程的影响，给出了根据异常处理策略列举执行流程的方法，建立了支持事务机制的组合服务描述模型。并基于该描述模型提出了组合服务执行期 QoS 属性均值预测算法。通过实验证明了在对具有事务机制的组合服务进行 QoS 预测时，本文方法的误差率明显低于未考虑事务机制的 workflow 预测方法，同时证明了本文方法具有较好的可行性和有效性。

参考文献

- [1] Zeng L, Benatallah B, and Ngu A H H, *et al.* QoS-aware middleware for web services composition. *IEEE Trans. on Software Engineering*, 2004, 30(5): 311–327.
- [2] Grassi V. Architecture-based dependability prediction for service-oriented computing. In: *Architecting Dependable Systems III* (R. de Lemos, A. Romanovsky, C. Gacek Eds.), 2005, LNCS 3549, 279–299.
- [3] Chanrdasekeran S. Composition, performance analysis and simulation of Web services. [Masters Theses], University of Georgia Athens, 2002.
- [4] Cardoso J. Quality of service and semantic composition of workflows. [PhD thesis], Department of Computer Science, University of Georgia, Athens, GA (USA), 2002.
- [5] Cardoso J, Sheth A, and Miller J, *et al.* Quality of service for workflows and web service processes. *Journal of Web Semantics*, 2004, 1(3): 281–308.
- [6] Jaeger M C, Rojec-Goldmann G, and Muhl G. QoS aggregation for service composition using workflow patterns. In *Proceedings of the 8th International Enterprise Distributed Object Computing Conference (EDOC 2004)*, Monterey, California, USA, September 2004, 149–159.
- [7] Jaeger M C, Rojec-Goldmann G, and Muhl G. QoS aggregation in Web service compositions. In *proceeding of the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service*, 29 March–1 April 2005, 181–185.
- [8] Neila Ben Lakhel, Takashi Kobayashi, and Haruo Yokota. A failure-aware model for estimating and analyzing the efficiency of Web services compositions. In *Proc. of IEEE 11th Intl Symposium on Pacific Rim Dependable Computing (PRDC2005)* Changsha, China, 12–14 Dec. 2005, 114–124.
- [9] Gray J and Reuter A. *Transaction Processing: Concepts and Techniques*. Morgan Kaufmann Publishers, San Francisco, Chapter 4: Transaction Models, California, 9th edition, 2002.
- [10] Younas M, Eaglestone B, and Holton R. A formal treatment of a SACReD protocol for multidatabase web transactions. *11th International Conference on Database and Expert Applications*, 2000, London, Springer LNCS, 899–908.
- [11] Zeng L, Lei H, Jeng J, Chung J, and Benatallah B. Policy-driven exception-management for composite web services. *Proceedings of the 7th IEEE International Conference on e-Commerce Technology (CEC'05)*, IEEE Computer Society, Munich, Germany, 2005: 355–363.
- [12] Papazoglou M P. Web services and business transactions. *World Wide Web*, 2003, 6(1): 49–91.
- [13] Pautasso C, Heinis T, and Alonso G. Autonomic execution of service compositions. In *3rd IEEE Int. Conf. on Web Services (ICWS'05)*, Orlando, USA, July 2005, 434–442
- [14] Pautasso C. A flexible system for visual service composition. [PhD thesis], ETH, July 2004.

吴江霞：女，1978年生，博士生，研究方向为新一代互联网服务体系、Web 服务组合。

杨放春：男，1957年生，教授，研究方向为新一代互联网服务体系。