

基于 WS-C/T 协议的 Web 服务业务事务处理研究与实现

何 演,管有庆

(南京邮电大学 物联网学院 信息网络技术研究所,江苏 南京 210003)

摘 要:与原子事务严格遵循 ACID 特性相比,Web 服务业务事务具有异构、分布式、长时间运行、松耦合的特点。它允许参与者独立提交,并使用补偿操作来撤销已提交的子事务所带来的影响。针对传统事务处理模型不适合处理业务事务的情形,设计一个基于 WS-C/T 协议的 Web 服务业务事务处理原型系统,把用于协调的 WS-C 协议和用于业务事务处理的 WS-BA 协议相结合来实现 Web 服务的组合应用。业务事务处理系统主要由客户端、协调者和参与者三个部分组成,客户端发起业务事务,协调者和参与者提供业务事务协调与处理接口,实现业务事务一致性。最后通过一个业务事务应用实例,验证了原型系统的可行性,它具有协调和补偿能力。

关键词:Web 服务业务事务;补偿;协调;WS-C/T 协议

中图分类号:TP393

文献标识码:A

文章编号:1673-629X(2011)04-0090-04

Research and Implementation of Web Services Business Activity Based on WS-C/T Protocol

HE Yan, GUAN You-qing

(Institute of Information Network Technology, College of the Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

Abstract: Compared with atomic transactions which comply with ACID properties, Web services business activity is heterogeneous, distributed, long running, loosely coupled. It allows participants to commit independently and undo committed sub-transactions by means of compensation. As traditional transaction mechanisms do not meet the requirements of business activity, a prototype system is designed of processing the Web services business activity based on WS-C/T protocol. To realize the application of Web services, it integrates WS-C protocol used in Web services coordination and WS-BA protocol used in processing of business activity. The system involves clients, coordinators and participants, clients initiating the business activity, the two others cooperating to provide the interfaces for coordination and process in order to realize the consistency of business activity. To verify the prototype system, an example is cited of Web services business activity with capability of compensation.

Key words: Web services business activity; compensation; coordination; WS-C/T protocol

0 引 言

随着 Internet 的蓬勃发展,越来越多的企业希望将自身的业务与服务通过互联网进行拓展。基于可扩展的标记语言(XML)的 Web Services 作为一种新型分布式计算技术,利用简单对象访问协议(SOAP)和 Web 服务描述语言(WSDL)等广为接受的标准协议可以方便地实现 Internet 上跨平台、松耦合的异构应用的交互和集成。在当前 Web 服务应用中,常常涉及多

个 Web 服务之间交互与协作的组合。与传统的分布式事务不同^[1],Web 服务面对的是一个广域分布的、异构与动态的环境,由于 Web 服务的调用是无状态的(即 Web 服务之间不存在会话),Web 服务事务协调处理就成了 Web 服务组合应用能否成功的关键。目前,比较常见的三个协议族:(1) WS-C/T^[2-4];(2) OASIS BTP^[5];(3) OASIS WS-CAF^[6]。都提出了在自治、异构的环境下如何建立可靠 Web 服务组合应用的事务处理框架。文献[7]使用两阶段提交协议保证事务的原子性和一致性,设计实现了一个 Web 服务事务处理系统 WSTPS;文献[8]基于 BTP 协议,设计实现了一个 Web 服务事务协调系统;文献[9]提出一种子事务延迟提交算法 STCD,用于实现 Web 服务业务

收稿日期:2010-08-28;修回日期:2010-12-04

基金项目:江苏省自然科学基金计划项目(05KJD520146)

作者简介:何 演(1985-),男,硕士研究生,研究方向为软件技术在通信网络中的应用;管有庆,副研究员,硕士生导师,研究方向为数据库、通信软件和下一代网络等。

事务(即长事务)失败时减少补偿活动的数目,降低补偿代价。文献[10]提出了一个 Web 服务容错模型 OpenWS-Transaction,通过日志技术来实现故障恢复机制,从而保证事务处理的正确性和可靠性。

1 Web 服务业务事务处理概念

1.1 WS-C 及 WS-BA 协议介绍

WS-C/WS-T(WS-Coordination/WS-Transaction)规范是由 IBM、Microsoft 于 2002 年 8 月发布的。WS-C 提供了一个协同 Web 服务达成事务一致性的框架,它允许应用程序协调一个活动中涉及的不同 Web 服务,它有三个关键元素:用于创建活动的激活服务、用于注册参与者的注册服务、用于活动完成处理的协调服务。它的核心概念是活动(Activity),一个活动实际上是一个计算过程,一般涉及多个 Web 服务,是一组 Web 服务操作的集合。

WS-BA(WS-Business Activity)是 WS-T 规范定义两种事务处理协议之一,它扩展了 WS-C 规范,主要用于处理长时间运行、松耦合的业务事务。业务事务中的任务一般由多个原子事务组成,如果整个事务失败,则由原子事务执行的对资源的更新操作都必须被撤销,补偿操作是处理这种情况的常用方法,因此该协议引入了补偿机制,要求事务的参与者必须自己实现补偿的逻辑,这样在事务被撤销时可以使用补偿逻辑来恢复数据的状态。WS-BA 支持两种模式^[4]:参与者提交结束模式和协调者提交结束模式,两者的区别在于参与者是否能主动地发送事务完成消息。

1.2 WS-BA 业务事务状态转换图

在业务事务环境中,各个参与者管理着自己的资源,这些资源通过预定义的行为方式协调起来。图 1 具体描述了 WS-BA 业务事务的协调者和参与者通过消息交互实现事务的状态转换。

图中的椭圆结点表示事务状态,它描述了发送消息者代表的事务所处的当前状态。协调者和参与者通过判断当前状态和发送的消息类型进行状态转换,其中虚线表示参与者产生的消息,实线表示协调者产生的消息,线上的注释表示了传递的消息。在整个协调过程中主要有 Active(活动状态)、Completing(提交中)、Completed(提交完成)、Canceling(取消任务中)、Exiting(退出任务中)、Closing(关闭中)、Compensating(补偿中)、Faulting(发送异常)、Ended(事务结束)等九种状态。

2 Web 服务业务事务处理实现

2.1 模型结构

Web 服务业务事务处理模型主要由客户端、协调者和参与者三个部分组成,模型结构如图 2 所示,其中协调者和参与者以 Web 服务形式部署在 Web 服务运行时容器中。

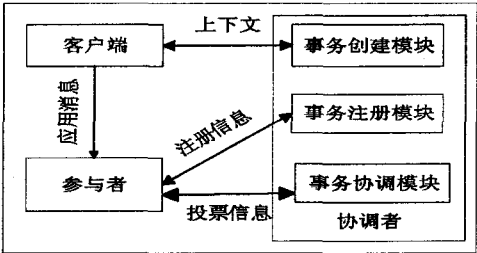


图 2 业务事务模型结构图

(1) 客户端:由客户端应用程序使用协调者提供的创建上下文函数请求创建事务。它可以通过接口函数传递具体的参数用以发起业务事务请求。

(2) 协调者:它管理和运行着整个事务,遵照 WS-C 规范来实现。

它包括三个模块:

1) 事务创建模块:用于接收客户端应用程序创建

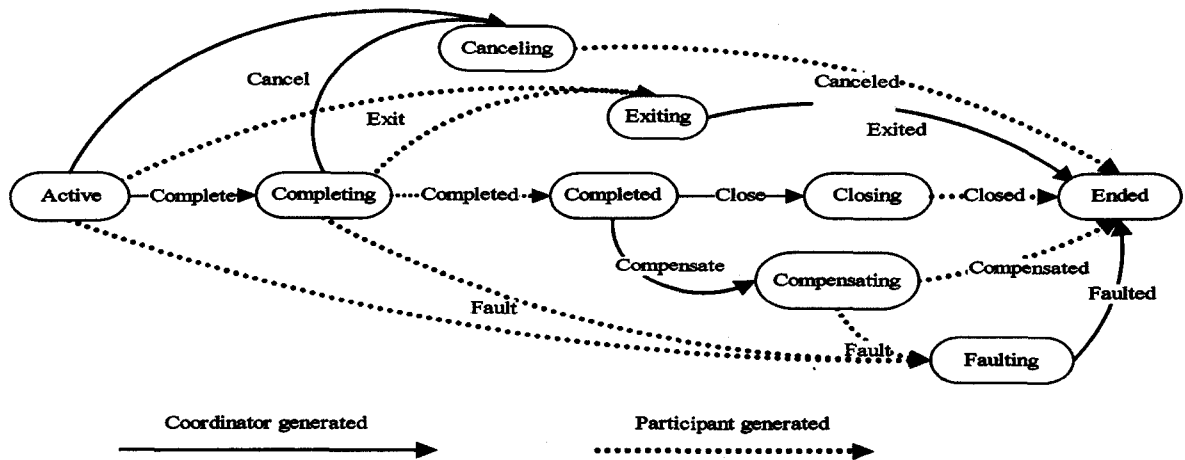


图 1 业务事务状态转换图

业务事务的请求,为新事务生成一个唯一事务 ID 号和相应的协调上下文,并将事务 ID 号和协调上下文关联起来,最后将它们返回到应用程序。

2) 参与者注册模块:用于将 Web 服务注册成为参与者。当被调用的 Web 服务收到协调上下文消息后,将注册成为一个事务的参与者。协调者根据注册消息的事务 ID 号,返回事务协调信息。

3) 事务协调模块:参与者根据应用程序发送的应用消息请求,得到一个处理结果,并将它作为子事务状态返回给协调器,最后由协调器根据所有子事务状态决定提交或补偿整个业务事务。

(3) 参与者:提供具体的服务。许多参与者一起完成 Web 服务的组合应用。

2.2 业务事务处理接口

业务事务的处理主要由两个部分组成:协调者接口

(Coordinator)和参与者接口 (Participant),它们的主要成员方法如图 3 所示。

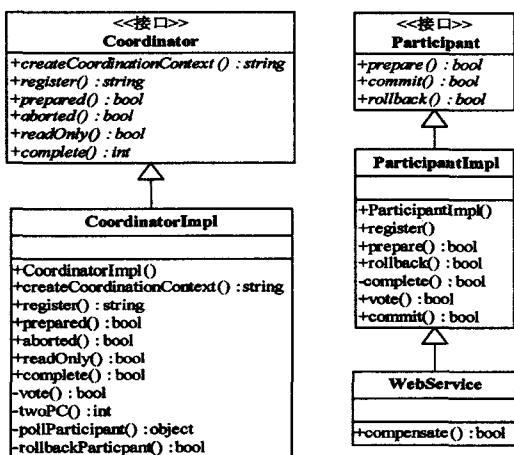


图 3 协调者接口 (左) 与参与者接口 (右)

1) 协调者的主要方法有:①createCoordinationContext() 创建协调者上下文,上下文主要包含 Identifier 事务标识符、Expires 活动的超时值、CoordinationType 协调类型等;②register() 注册参与者;③prepared() 判断参与者是否已经准备好提交事务;④aborted() 判断参与者是否回滚事务;⑤complete() 指示所有参与者提交事务。

2) 参与者的主要方法有:①prepare() 准备好事务;②commit() 提交事务;③rollback() 回滚事务;④vote() 投票,参与者准备好了投“commit”,没准备好或

发生异常则投“rollback”。参与者最重要的方法补偿 compensate() 一般在继承于 ParticipantImpl 类的具体服务中根据业务逻辑来实现。

2.3 业务事务处理流程

业务事务处理通常需要协调者与多个参与者交互作用来完成,其基本处理流程如图 4 所示。

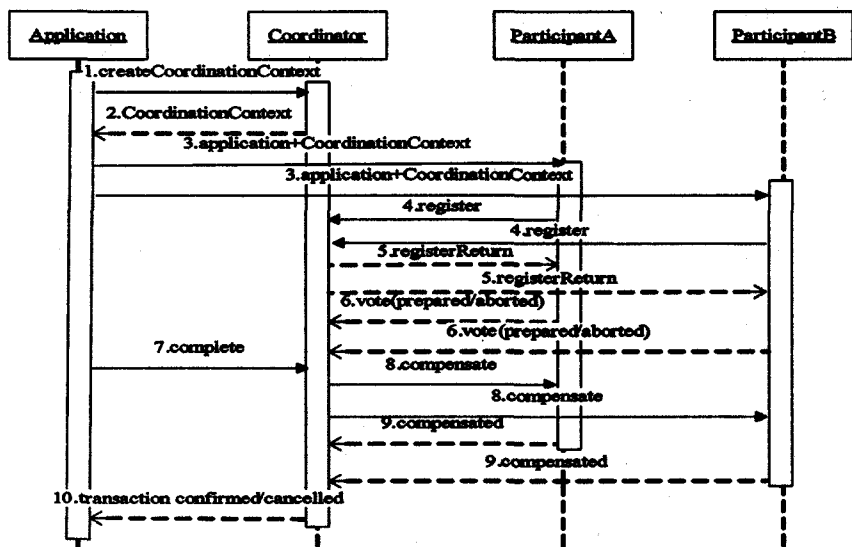


图 4 业务事务处理流程

图中的执行过程如下:1) 应用程序发送创建上下文消息给协调者激活事务。2) 协调者返回事务上下文的消息给应用程序。3) 应用程序发送服务请求和上下文给参与者。4) 参与者向协调者发送注册请求。5) 协调者向参与者返回已注册消息,表示已成功加入事务。6) 参与者向协调者发送服务请求的处理投票 (vote) 结果,可能成功了 (prepared),也可能失败 (aborted)。7) 应用程序向协调者发送 complete 请求,目的为获得整个业务事务的最终结果。8) 和 9) 为可选步骤,如果所有参与者处理投票的返回结果都为成功 (prepared),则可以跳过,否则参与者需补偿先前操作。10) 协调者返回最终结果给应用程序,confirmed 表示成功,cancelled 表示失败。

3 业务事务应用实例

旅行社代理系统是一个典型的需要保障可靠性的 Web 服务业务事务。本应用实例将松散的业务 (预定航班和预定旅馆) 集合到一起形成一个组合事务,为了保证整个事务流程的可靠性,它必须能够协调各个服务的活动。如果其中一个服务操作失败,那么整个旅行社代理服务也将取消,同时要对已提交到后台数据库中的操作,如 update, insert 等进行逆向补偿。本实例在 OpenWS-Transaction 基础上,通过自定义的数据库函数来实现业务的补偿操作。本例使用 Tomcat 5.5、Axis 1.4、JDK 1.5 作为基础平台,MySQL 作为后台

```
Participant http://localhost:8080/WSCT1.1/services/FlightServiceLR voted XA_OK on 'reserveFlight'
Participant http://localhost:8080/WSCT1.1/services/HotelServiceLR voted XA_OK on 'reserveHotel'
Transaction is COMMITTING
Transaction is COMMITTED
1) Transaction protocol finished for transactionID 192.168.65.1/2010-08-15|21:34:13.703
```

图5 事务运行成功

```
Participant http://localhost:8080/WSCT1.1/services/HotelServiceLR voted XA_OK on 'reserveHotel'
Participant http://localhost:8080/WSCT1.1/services/FlightServiceLR voted XA_ROLLBACK on 'reserveFlight'
Transaction is ROLLING BACK
Room Reservation be compensated successfully
Transaction is ROLLED BACK
1) Transaction protocol finished for transactionID 192.168.65.1/2010-08-15|21:07:15.421
```

图6 事务运行失败

数据库。其中 Tomcat 是一个 Web 服务器, Axis 是一个基于 SOAP1.2 规范的 SOAP 引擎。Handler 和 Chain 是 Axis 引擎提供的服务流程处理机制。将每个服务都具有的功能(如注册、投票)设计为 Handler 可以减少代码的冗余,将请求响应流程设计为 Chain 形式,可以降低处理流程的耦合度。

实例部分代码:

//旅行社代理的协调者继承于实现了 Coordinator 接口的 CoordinatorImpl 类

```
public class TravelCoordinator extends CoordinatorImpl {
```

//返回协调者上下文

```
public String createCoordinationContext (.....) throws CoordinationException {.....} }
```

//航班服务继承于实现了 Participant 接口的 ParticipantImpl 类

```
public class FlightService extends ParticipantImpl {
```

```
public void reserveFlight (.....) {} //预定机票
```

```
public boolean compensate (.....) {
```

①连接数据库;②逆向补偿操作(采用存储过程);③关闭连接;
④返回“补偿”操作成功与否。

```
}
```

//旅馆服务继承于实现了 Participant 接口的 ParticipantImpl 类

```
public class HotelService extends ParticipantImpl {
```

```
public void reserveHotel (.....) {} //预定旅馆
```

```
public boolean compensate (.....) { } //补偿操作
```

```
}
```

业务事务实例运行成功或失败的结果如图 5、6 所示。

从图 5 中可以看出参与者旅馆服务、航班服务都投了 XA_OK(在 J2EE 标准中^[11]表示两阶段提交事务成功),当应用程序发出 complete 指令后,所有参与者都提交事务,整个事务成功。从图 6 中可以看出参与者旅馆服务投了 XA_OK,而航班服务投了 XA_ROLLBACK(失败),当应用程序发出 complete 指令后,参与者旅馆服务进行要 compensate 补偿操作,整个事务失败。

以上两图说明组成一个事务的预定航班、预定旅馆两个服务,当它们都成功时,整个事务成功,若任何一个服务失败,则其它已提交成功的服务要进行逆向

补偿操作,整个事务失败。

4 结束语

Web 服务组合是业务事务应用中的常见形式,为了保证组合的 Web 服务正确执行和得到一致结果,必须为组合的 Web 服务提供事务协调与补偿机制。文中在研究了相关 Web 服务事务标准的基础之上,利用 Axis、Tomcat 等开放源码工具,实现了一个基于 WS-BA 规范的 Web 服务业务事务处理系统,为 Web 服务组合提供了事务处理的支持,确保了组合服务的结果一致性。目前,Web 服务业务事务的补偿机制主要是基于关系型数据库事务的自动补偿方法^[12],因此后续工作需要考虑在执行补偿事务遭遇失败时故障恢复的问题。

参考文献:

- [1] 王晓东,姜 浩. Web Service 同传统分布式技术的比较分析[J]. 计算机技术与发展,2008,18(3):125-131.
- [2] BEA,IBM,Microsoft. Web Services Coordination (WS-Coordination) [EB/OL]. 2005-08. <http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-tx/WS-Coordination.pdf>.
- [3] BEA,IBM,Microsoft. Web services atomic transactions (WS-AtomicTransaction) [EB/OL]. 2005-08. <http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-tx/WS-AtomicTransaction.pdf>.
- [4] BEA,IBM,Microsoft. Web Services Business Activity Framework (WS-BusinessActivity) [EB/OL]. 2005-08. <http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-tx/WS-BusinessActivity.pdf>.
- [5] OASIS BTP Committee. Business Transaction Protocol Specification [EB/OL]. 2002-06-03. http://www.oasis-open.org/committees/download.php/1184/2002-06-03.BTP-cttee_spec_1.0.pdf.
- [6] Bunting D, Chapman M, Hurley C, et al. Web Services Composite Application Framework [EB/OL]. 2005-10. http://www.oasis-open.org/committees/documents.php?wg_abb

(下转第 97 页)

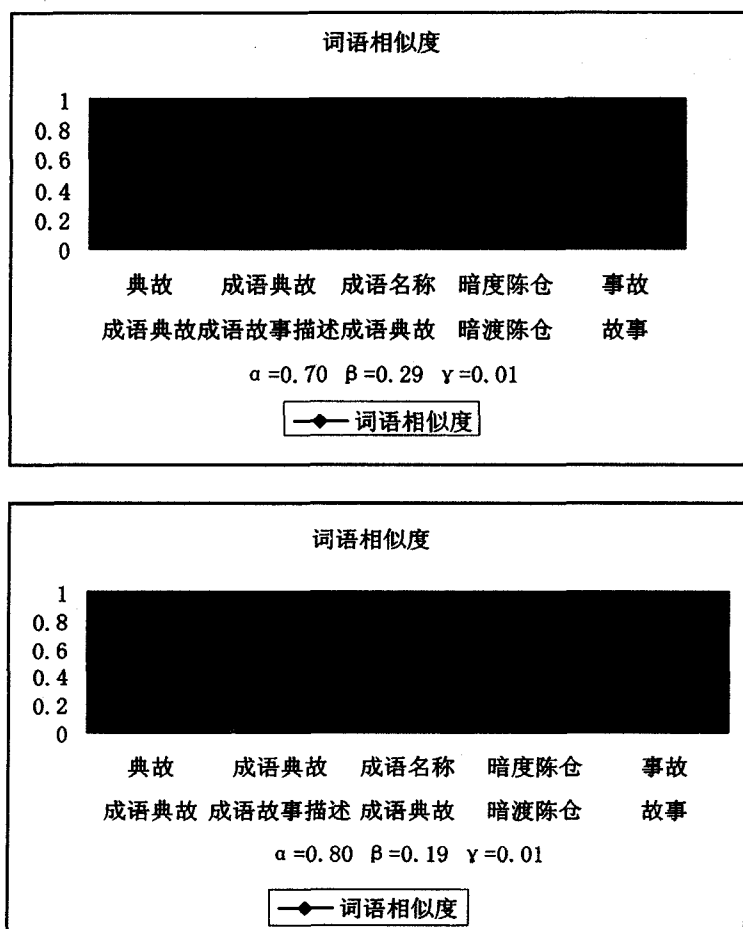


图3 不同概念对在两组参数取值下
相似度计算结果对照图表

息检索领域有着重要的作用。文中建立了基于成语典故本体的词语相似度计算方法,分别从语素、字序和词长三方面来计算词语的相似度,并对方法中的参数取值进行了详细的讨论,通过实验对计算方法进行了验证。从实验结果上看,计算方法的结果与常识判断基本一致。由于本体库中的词汇较多,在测试时,只计算了部分概念同用户输入词汇的概念相似度,另外,用户输入词汇也是主观判定;其次,本体库中相关关系含义广泛,文中在处理中没有进行区分,这有可能影响到计算结果的精度。这将是下一步研究的工作。

(上接第93页)

rev=ws-caf.

- [7] 尹 瑞,葛 声,马殿富. Web 服务事务处理系统的研究与实现[J]. 北京航空航天大学学报,2005,31(4):439-442.
- [8] 张小欢,管有庆,王佳伟. 基于 BTP 的 Web 服务事务协调研究与实现[J]. 计算机技术与发展,2010,20(4):39-42.
- [9] 朱 锐,郭长国,王怀民. 一种基于补偿代价的长事务调度算法[J]. 软件学报,2009,20(3):744-753.
- [10] Vasquez I, Miller J, Verma K. OpenWS-Transaction: Enabling

参考文献:

- [1] 秦春秀,赵捧未,刘怀亮. 词语相似度计算研究[J]. 情报理论与实践,2007,30(1):105-108.
- [2] 章志凌,虞立群,陈奕秋,等. 基于 Corpus 为库的词语相似度计算方法[J]. 计算机应用,2006,26(3):638-644.
- [3] 赖院根,王 娜. 概念语义相似度计算与参数估计[J]. 情报杂志,2009,28(8):148-152.
- [4] 康文宇,杨志强. 相似度计算在智能答疑系统中的研究及应用[J]. 计算机技术与发展,2010,20(2):63-66.
- [5] Gruber T R. A translation approach to protable ontology specifications, KSL 92-71 [R]. [s. l.]: Knowledge System Laboratory, 1993.
- [6] Richardson S D, Dolan W B, Vanderwende L. Acquiring and structuring semantic information from text [C]//ACL '98 Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2. [s. l.]: [s. n.], 1998: 1098-1102.
- [7] Li Sujianl. Semantic computation in Chinese question-answer system [J]. Journal of Computer Science and Technology, 2002, 17(6):933-939.
- [8] 赵 军,胡栓柱,樊兴华. 一种新的词语相似度计算方法[J]. 重庆邮电大学学报(自然科学版),2009,21(4):528-532.
- [9] 朱毅华,侯汉清,沙印亭. 计算机识别汉语同义词的两种算法比较和测评[J]. 中国图书馆学报,2002(4):82-85.
- [10] 吕学强,任飞亮,黄志丹,等. 句子相似模型和最相似句子查找算法[J]. 东北大学学报(自然科学版),2003,24(6):531-534.
- [11] 周法国,杨炳儒. 句子相似度计算新方法及其在问答系统中的应用[J]. 计算机工程与应用,2008,44(1):165-167.
- [12] 冉 婕,孙 瑜,昌 霞,等. 基于 OWL 的成语典故本体构建研究[J]. 计算机技术与发展,2010,20(5):63-66.

Reliable Web Service Transactions [EB/OL]. 2005-09. <http://lsdis.cs.uga.edu>.

[11] Oracle. Java EE 5 Technologies [EB/OL]. 2010-06. <http://java.sun.com/javaee/technologies/javaee5.jsp>.

[12] 唐飞龙,李明禄,曹 健. 一个 Web 服务事务处理模型:结构、算法和事务补偿[J]. 电子学报,2003,31(12):2074-2078.