

基于 REST 架构的小额支付平台研究

邓玉龙^{1,2,3}, 张 岩¹, 马玉鹏¹

(1 中国科学院 新疆理化技术研究所, 新疆 乌鲁木齐 830011; 2 新疆民族语音语言信息处理实验室, 新疆 乌鲁木齐 830011; 3 中国科学院大学, 北京 100049)

摘 要: 针对现阶段小额支付平台终端繁杂异构的问题, 提出了一种基于 REST 架构风格的小额支付系统平台, 其将小额支付过程以及相关数据抽象为资源, 以 HTTP 作为通信协议, 通过 RESTful Web 服务方式连接移动客户端与 Web 服务端, 设计了小额支付平台的架构, 定义了小额支付平台 RESTful Web 服务的具体方法和规则, 并且实现了小额支付原型系统. 实验结果表明, 相较于传统的基于 SOAP 协议的大 Web 服务方式, 基于 REST 架构风格的小额支付平台具有明显的优势.

关键词: 小额支付; REST 架构; Web 服务; SOAP

中图分类号: TP399

文献标识码: A

文章编号: 1000-7180(2017)10-0086-05

Research on Micro Payment Platform Based on REST Architecture

DENG Yu-long^{1,2,3}, ZHANG Yan¹, MA Yu-peng¹

(1 The Xinjiang Technical Institute of Physics & Chemistry, Chinese Academy of Sciences, Urumqi 830011, China;
2 Xinjiang Laboratory of Minority Speech & Language Information Processing, Chinese Academy of Sciences
Urumqi 830011, China; 3 University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Aiming at the terminal complex heterogeneous problems of payment platform, a kind of REST architecture based on the micro payment system platform is proposed. The micro payment process and related data are described as resources. HTTP is used for the communication protocol. Mobile client and the Web server communicates with RESTful Web service. The architecture of the micro payment platform was designed, and the specific methods and rules of the micro payment platform RESTful Web service are defined. The experimental results show that compared with the traditional Web based SOAP services, micro payment platform based on the REST architecture has obvious advantages.

Key words: micro payment; REST architecture; Web Service; SOAP

1 引言

随着移动互联网相关技术的飞速发展, 以手机、PDA 等移动终端为工具, 通过移动通信网络实现的移动小额支付^[1] (小于 100 元) 方式越来越受到大家的喜爱. 移动小额支付以其便携性、实时性, 突破地域、时间限制, 为人们提供安全、便捷、灵活的交易体验, 成为互联网支付发展的重要方向. 然而随着网络

用户的暴增、移动支付业务的大量涌现, 以基于 SOAP 协议的大 Web 服务为代表的系统^[2-5] 在集成过程中逐渐出现了一些问题: 系统逐渐设计得越来越臃肿, 在面临多用户、高并发、分布异构^[6] 的场景时, 系统逐渐力不从心, 用户感知性能有了明显的下降; 系统耦合性过高, 导致系统可伸缩性比较差, 系统升级改造变成了一件庞大而复杂的过程.

本文提出了基于 REST 架构的小额支付平台,

收稿日期: 2017-01-09; **修回日期:** 2017-02-20

基金项目: 中国科学院西部之光人才培养计划项目(XBBS201319); 中国科学院青年创新促进会资助项目; 新疆维吾尔自治区引进高层次人才计划资助项目

将支付过程以及支付过程中所涉及到的数据抽象为资源,使得系统各异构模块采用 RESTful Web Service 方式进行组建,设计了系统架构,定义了小额支付 RESTful Web 服务的具体方法和规则。

2 大 Web 服务与轻量级 REST 服务

传统的基于 SOAP 协议的 Web 服务搭建的平台,其将应用抽象为一组服务过程,使用 WSDL 对相关服务进行描述,并且在 UDDI 上进行注册供用户进行查询,客户端通过 SOAP 协议调用相应的服务过程,目前大多数 Web 服务都是基于 WS- $\ast$ 协议栈构建的,它是 RPC 风格的重量级实现。然而,随着相应的 Web 服务的用户数量的增加、系统功能的复杂性变大等一系列问题的出现,系统规模越来越大,大 Web 服务的缺陷逐渐暴露出来了。大 Web 服务需要在客户端和服务端解析和封装 SOAP 数据包,消耗了系统性能,传输负载也会减少,并且将 SOAP 作为通信协议,间接放弃了 Internet/WWW 提供的服务器、网关、代理、客户端等简单的基础设施,丧失了直接使用 HTTP 协议的优势,没有合理地利用其缓存机制,降低了系统的可伸缩性;大 Web 服务是一种 RPC 风格的接口,这种接口需要开发人员负责生成、实现和维护,一旦系统需要添加新的功能,那么就需要增加新的接口,并改变接口的内部实现,这在很大程度上增加了开发难度,除此之外,当服务提供者改变特定接口往往导致服务的中断或者不可用^[7],这无疑导致了服务双方产生了紧耦合。

REST^[8]架构是由参与 HTTP 协议设计的 Roy Feilding 博士提出的,该架构风格所有资源都有一个 URI,客户端使用 HTTP 协议来访问资源,通过 HTTP 中的 GET、POST 等方法对资源做不同的操作,实现了接口的统一。客户与请求服务器之间可通过 JavaScript Object Notation (JSON) 进行数据传输。这种交互模型同样可以适用于 Web 服务,每一个 Web 服务可以通过 URI 标识,我们通过 HTTP 所提供的通用简洁的接口来访问服务。REST 架构的主要特征如下。

可寻址性:架构中的每个资源或者对象都可以通过一个唯一的 URI 进行寻址,URI 的结构一般比较简单,和数据资源有一定的层次关联,便于理解和使用。

无状态性:服务器可能的状态即就是所有的资源,服务端不会保留客户端的状态。

连通性:客户端应用状态应该在超媒体的指引下转移,即就是所谓的表述性状态移交 (REST)。

统一接口:REST 中的所有动作和访问的资源都可以通过 HTTP 请求得到,而且方法必须使用 HTTP 规定好的方法,包括 POST (增)、DELETE (删)、PUT (改)、GET (查)、HEAD、OPTIONS、PATCH、LINK、UNLINK。一般常用的是前四种,完成项目中常用的 CURD 操作。

3 基于 REST 架构的小额支付平台

考虑到小额支付系统的性能、可用性、安全性等指标,我们设计了小额支付平台的部署架构图,如图 1 所示。

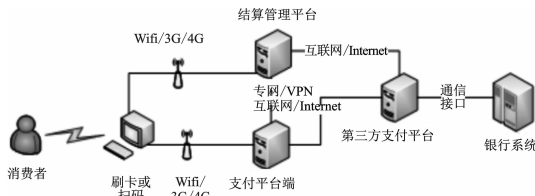


图 1 小额支付平台的部署架构图

小额支付涉及的主要部分的如下。

(1)消费者:这里指的是在商户或者业务终端采用移动小额支付手段进行消费的顾客。

(2)结算平台:结算平台管理商户支付的必要信息,同时在一定的周期内完成对商户的交易的扎差结算功能。

(3)第三方支付平台:第三方支付平台与银行系统签订了金融合作协议,银行系统委托第三方支付完成一系列网络运营和金融服务,这里主要指负责对用户(消费者、商户、结算方)的账户信息进行管理,同时完成账户的支付功能,本文中涉及到的第三方支付包括 POS 支付、支付宝支付、微信支付。

(4)商户业务终端:业务终端指的自动售货机、超市、农贸市场充电桩等适应小额支付手段的商户,它提供给消费者便捷的小额支付服务。

(5)小额支付平台:小额支付平台作为整个系统的核心,它是连接了消费者、业务终端、第三方支付服务平台,主要负责商业用户的注册、身份验证、绑定,商户各类资源管理;集成第三方支付平台,为商户端 APP 提供便捷、灵活的支付接口,同时对 App 版本进行管理;完成与结算平台的交易核算、结算业务;负责对商户的交易、结算数据的统计分析管理。

3.1 小额支付资源模型建立

REST 是一种面向资源的服务架构,那么什么样事物可以被认为是资源呢?就其本质而言,任何足够重要并且被其他事物引用的事物都可以视为,《万维网的架构》一书中指出:“想要建立指向某个事物的超文本链接,提出或者反对关于它的断言,获取或者缓存它的表述,供其他表述引用它的全部或者部分,给他增加注释信息,或者对它执行某些操作.”这时,你都应该将它定义为资源.

所以在设计 REST 架构风格的小额支付平台时,首先需要从业务中进行资源的抽象出资源,本文将根据小额支付相关的业务特点,进行资源的设计,其中包含对资源的名称的设计,使用 URI 进行标识.

本文中涉及的业务主要包含各类商户、商户下属的门店、终端 POS 机,APP 上用户的注册、登录鉴权管理,商户的支付过程以及随之而来的交易数据和结算数据的管理,除此之外商户还有一些相关的管理部门数据、系统参数等,这些业务围绕的资源大致可以分为商户及其下属部门(门店、POS 机)数据、交易数据、结算数据、发卡商、机构数据、用户数据以及用户订单、系统参数.其中,用户订单按照支付方式分为三种方式:POS 支付方式、微信支付、支付宝支付.

如下为查询所有商户信息的 URI 模板:

http://../merchants/? start = {start} &-size = {size} &-sort = {sort}

在该 URI 中 merchants 代表的是所有商户, start 和 size 完成商户数据的分页, sort 则用于进行排序方式的选择. 商户信息相关 RESTful 接口如表 1 所示.

表 1 商户信息相关 RESTful 接口

接口描述	资源请求 URL	请求方式
查询所有商户 (分页/排序)	http://../merchants/? start = &-size = &-sort =	GET
查询某种条件的商户	http://../merchants/{ condition }	GET
某个 id 的商户	http://../merchants/merchant? no =	GET、DELETE、PUT、POST
查询某种条件的商户 (分页/排序)	http://../merchants/{ condition }? start = &-size = &-sort =	GET
查询某种条件的商户	http://../merchants/{ condition }	GET

3.2 媒体类型选择与设计

传统的 RPC 风格 Web service 的设计仅仅将

XML 格式作为传输媒体的唯一选择,而随着系统的复杂以及用户量的爆炸式增长,XML 媒体格式逐渐显现出它本身的弱点,过于臃肿,解析复杂,这样也就造成了系统的性能不佳. 基于 REST 架构风格的 Web Service,秉承 HTTP 协议的优点,支持多种媒体格式,常见的有 XML、JSON、HTML、Atom、XHTML 等,这些超媒体格式还可以进行一定的扩展与组合,比如: collection + JSON、Maze + XML 等,本文中选用更轻、更友好的媒体类型 JSON 作为数据传输的格式. 比如:获取一个商户的个人信息数据,我们就可以通过 HTTP 协议,发送一个这样的请求:GET http://localhost:8080/xezf/merchants/,那么就会返回到如下的 JSON 格式的数据内容:

```
[
{ "merchantName": "刘 XX",
  "merchantNo": "999290001000147",
  "area": "上海市",
  "entityName": "XX 菜市场",
  "address": "XX 路 XX 号",
  "telephone": "138XXXXXXXX",
  "bank": "中国银行",
  "account": "XXXXXXX",
  "payeeName": "刘 XX",
  "cycle": "月结算" }
```

查看表 1 可以知道,该资源还支持 DELETE、PUT、POST 等操作,那么能够以 POST 方式向路径 http://localhost:8080/xezf/merchants/999290001000147 发送携带内容为[{"cycle": "日结算"}] 的请求,最后就会实现将商户号为 999290001000147 的商户的结算周期从月结算改为日结算.

3.3 小额支付业务流程

小额支付平台的支付方式分为 POS 支付、微信支付、支付宝支付,其交易可过程抽象为图 2 所示的过程.

交易开始之前商户需要先进行注册,本文中小额支付平台注册一律在 APP 端进行注册,商户手机安装该 APP 后,填写必要信息,进行注册.

然后,需要结算平台对商户进行身份审核,一旦审核通过,该商户就成为了小额支付平台的正式用户,可以进行一系列的交易活动.

此处 REST 架构风格的连通性(即客户端应用

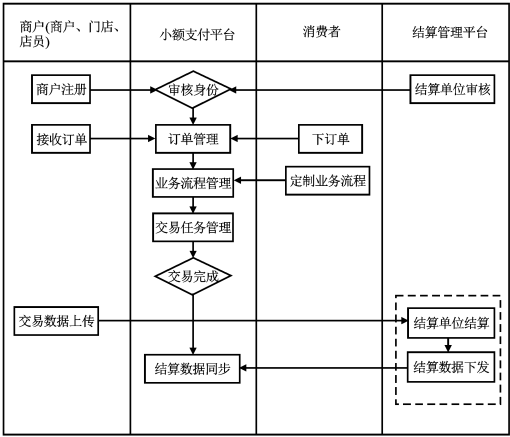


图 2 支付交易流程图

状态应该在超媒体的指引下 进行转移)发挥了作用, REST 架构风格中将 HTTP 作为一种应用协议,将超媒体作为一种应用状态的引擎 (HATEOAS), APP 客户端在超媒体的指引下,有序地推进当前的工作流程,引导 APP 发送各种适合当前场景的请求.

支付过程如下:

- (1)商业用户选择支付方式,选择小额支付平台支持任意一种,并填写金额和选择相应的交易类型等信息,紧接着调用相应的 RESTful 支付请求接口;
- (2)消费者通过手机端应用(支付宝 APP,微信 APP)扫码支付,同时查看支付金额等支付信息,确认完毕之后提交支付确认请求,等待支付响应的返回.
- (3)接下来,进行的操作,需要考虑用户需要而考虑进行响应的操作,比如,由于第三方支付系统或者小额支付平台出现网络或者硬件问题,响应比较慢,用户可以根据需要进行进行订单取消或者选择继续等待,直到第三方支付成功信息返回. 商户 APP 查询交易成功与否信息通过小额支付平台封装的查询接口进行查询,获取相关状态信息,如果交易成功返回响应的成功信息,如果失败小额支付平台调用订单撤销接口完成订单撤销. 消费者的交易成功与否信息由消费者 APP 自行维护.

4 原型系统与性能评估

4.1 原型系统搭建

按照图 1 所示的平台部署拓扑图,本文采用 JSR311 规范的参考实现框架 Jersey 对项目进行了原型实现,包含用户登录、注册、支付、交易查询、结

算查询等功能. 如图 2 所示,其中图 3(a)展示了用户 APP 登录界面,这里支持在图 3(b)已经注册过以及在支付平台登录的用户进行登录. 图 3(c)展示了支付宝支付界面,这里做测试一般选用 0.01 元作为交易额. 图 3(d)为历史结算信息查询结果界面,一般选用 N+1 结算规则进行结算. 图 3(e)展示了当天的交易流水信息,这里为了流水信息的实时显示,采用了本地存储的方式,其中可以看到采用了何种支付方式进行支付,图中展示了微信支付、银联 POS 支付以及支付宝支付的流水信息. 其中涉及到的硬件资源包括 4 台阿里云服务器、4 部手机、1 个 MPOS 机.



图 3 小额支付移动业务终端界面

4.2 基于 SOAP 协议的大 Web 服务与基于 REST 架构的 Web 服务测试

这一部分,比较基于 REST 架构的小额支付平台和其他架构类型的小额支付平台的性能,从终端或者用户的不同方面进行比较,收集和分析平台的性能表现,评估基于 REST 架构的小额支付平台. 由于基于 SOAP 协议的大 Web 服务标准是基于 XML 文件格式进行数据交换的,而 REST 架构更倾向于采用更加简洁的 JSON 格式进行数据交换的. 这里对 JSON 和 XML 的封装和解析时间性能进行测试,数据来源于小额支付平台前期积累是交易数据. 使用了 Gson 框架对交易数据进行了 JSON 的封装和解析,对于 XML 的封装和解析使用了 DOM4J 中的 SAX 方法进行了封装和解析.

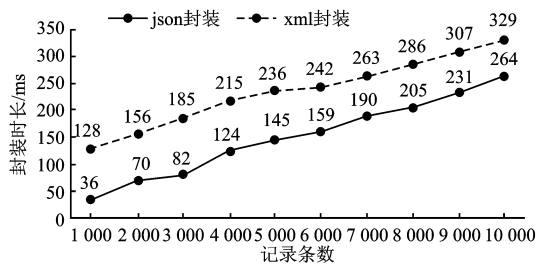


图4 JSON和XML封装时长图

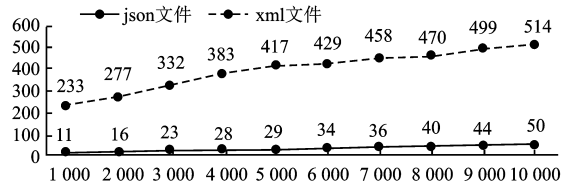


图5 JSON和XML解析时长图

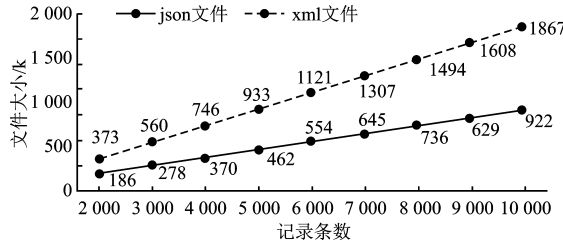


图6 JSON和XML文件大小

由图4可以看出,xml封装时间始终高于JSON封装的时间,而且随着记录条数的增加,二者的封装时间差距依然很大;由图5可以看出,xml解析时间随着记录条数的增加迅速增长,而JSON格式的数据在记录条数迅速增大时,解析时间几乎为常数;由图6可以看出,JSON格式的文件大小在基数上小于XML文件,而且随着记录条数的增加XML文件大小迅速增大。综上可以得出,XML在数据交互过程中,性能与JSON格式有比较大的差距。

在大Web服务架构和REST架构下对同一笔当日交易数据分别进行20次的查询,测试二者接口的时间性能,最终得出HTTP作为协议时访问耗时为1149.11ms,SOAP作为协议时耗时为2688.44ms(如图7所示),后者其通信时长是前者的2.3倍以上,由此可以得出,相较于将SOAP作为通信协议大Web服务,REST风格直接使用HTTP作为通信协议,数据传输更加简单快速。

5 结束语

结合目前小额支付平台架构的研究现状,分析传

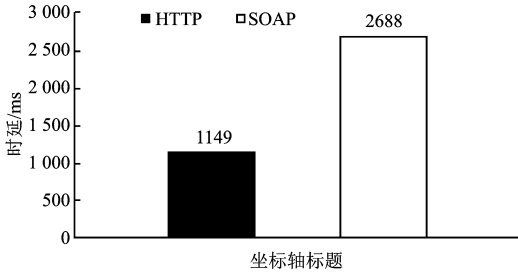


图7 REST风格服务和大Web服务数据访问时延

统的Web Service架构风格,研究了出基于REST的小额支付交易平台;使用Jersey技术框架进行平台设计,与传统的小额支付平台比较,该平台具有用户感知性能好、扩展性能高、可靠性强等特点,具有兼容自动终端、第三方支付等多种支付手段,以及用户体验好、交付效率高、系统运营成本低、安全性好、升级简单方便等实际效益,该平台适用于零售、餐饮、公共交通等行业,可支持常见的刷卡、扫码等快捷支付手段。

目前,基于REST架构风格的小额支付平台尚处于原型开发阶段,其中还有很多需求需要完成以及性能需要提高,例如小额支付平台在语义表示方面有所欠缺,在超媒体的状态转移表达方面实现不足。

参考文献:

[1] 刘丹,房宪鹏.移动支付国内外研究综述[J].商业时代,2009,28(25):106-107.

[2] Huang Xiaodi , Bao Jinsong , Dai Xiaoling , et al. M&E-NetPay: A micropayment system for mobile and electronic commerce[J]. symmetry ,2016, 8(8):74.

[3] Tewari H,O'Mahony D. Multiparty micropayment for Ad Hoc network[C]// Proceedings of the Wireless-Communications and Networking, 2003 (WCNC 2003). New Orleans, LA, USA, 2003 :2033-2044.

[4] Ziang N, Liu X, Zhao J,et. al. A mobile micropayment protocol based on chaos[C]// Proceedings of the 2009 Eighth International Conference on Mobile Business. Dalian, China, 2009: 284-289.

[5] Dai X, Grundy J. Net Pay: An off-line, decentralized micro-payment system for thin-client applications. Electron. Commer[J]. Electronic Commerce Research & Applications, 2007, 6(1):91-101.

[6] 陈赞,胡坚,汪超亮,等.基于REST Web Services的分布式遥感数据检索技术研究[J].计算机工程与设计,2010,31(14):3207-3209.

[7] 覃国蓉,何涛.REST在异构移动客户端接入中的应用研究[J].计算机应用与软件,2015,31(4):77-79.

参考文献:

- [1] Jung T, Li X Y, Wan Z, et al. Privacy preserving cloud data access with multi-authorities[C]// Proc. of IEEE INFOCOM, IEEE, 2013: 2625-2633.
- [2] Yu S, Wang C, Ren K, et al. Achieving secure, scalable, and fine-grained data access control in cloud computing[C]// Proc. of IEEE Conference on Information Communications, IEEE, 2010: 534-542.
- [3] Yu S, Wang C, Ren K, et al. Attribute based data sharing with attribute revocation[C]// Proc. of ACM Symposium on Information, Computer and Communications Security, ASIACCS 2010, Beijing, China, 2010: 261-270.
- [4] Matsuda T, Nishimaki R, Tanaka K. CCA proxy re-encryption without bilinear maps in the standard model [C]// Proc. of International Conference on Practice and Theory in Public Key Cryptography. Springer—Verlag, 2010: 261-278.
- [5] Frikken K, Atallah M, Li J. Attribute-based access control with hidden policies and hidden credentials[J]. IEEE Transactions on Computers, 2012, 55 (10): 1259-1270.
- [6] Zhang Y, Chen X, Li J, et al. Anonymous attribute-based encryption supporting efficient decryption test [C]// Proc. of ACM Sigsac Symposium on Information, Computer and Communications Security. ACM, 2013: 511-516.
- [7] Narayan S, Safavi-Naini R. Privacy preserving EHR system using attribute-based infrastructure [C] // Proc. of ACM Cloud Computing Security Workshop, Csw 2010, Chicago, 2010: 47-52.
- [8] Ruj S, Nayak A, Stojmenovic I. DACC: Distributed access control in clouds[C]// Proc. of IEEE International Conference on Trust, Security and Privacy in Computing and Communications, IEEE, 2011: 91-98.
- [9] Benaloh J, Chase M, Horvitz E, et al. Patient controlled encryption: ensuring privacy of electronic medical records[C]// Proc. of ACM Cloud Computing Security Workshop, Csw 2009, Chicago, IL, USA, November. DBLP, 2009: 103-114.
- [10] Vimercati S D C D, Foresti S, Jajodia S, et al. Over-encryption: management of access control evolution on outsourced data[C]// Proc. of International Conference on Very Large Data Bases. VLDB Endowment, 2007: 123-134.
- [11] Hur J, Noh D K. Attribute-based access control with efficient revocation in data outsourcing systems[J]. IEEE Transactions on Parallel & Distributed Systems, 2011, 22(7): 1214-1221.

作者简介:

蔡炎伟 男, (1991-), 硕士研究生. 研究方向为云计算安全. E-mail: 1228338405@qq.com.

白光伟 男, (1961-), 博士, 教授, 博士生导师. 研究方向为无线传感器网络、移动互联网、网络体系结构和协议、网络系统性能分析和评价、多媒体网络服务质量等.

沈航 男, (1984-), 博士, 讲师. 研究方向为云计算、移动互联网、无线多媒体通信协议等.

朱德剑 男, (1991-), 硕士研究生. 研究方向为移动云计算、云计算节能技术.

(上接第 90 页)

- [8] Roy T Fielding, Richard N Taylor. Principled design of the modern Web architecture[D]. Irvine: Information and Computer Science University of California, 2000.

作者简介:

邓玉龙 男, (1990-), 硕士研究生. 研究方向为物联网技术、大数据分析. E-mail: 379879419@qq.com.

张岩 男, (1962-), 研究员. 研究方向为自动控制.

马玉鹏 男, (1979-), 副研究员. 研究方向为物联网应用技术、大数据分析.