

Using Web Services in the Internet Banking Transaction System

Phung Huu Phu and Myeongjae Yi

School of Computer Engineering and Information Technology, University of Ulsan

San 29, Mugeo-2, Dong Nam-Gu, 680-749 Ulsan, Korea

e-mail : {phungphu, ymj}@mail.ulsan.ac.kr

Abstract

Web Services are rapidly becoming the enabling technology of today's e-business and e-commerce systems, and will soon transform the Web into distributed applications. The aim of this paper is to present our approach in applying Web Services into the Internet transaction banking system. The system is integrated among backend system of banks so that each bank system can communicate with each other in order to perform transactions in transferring money from an account in one bank to an account in the partner bank and vice versa. The backend systems of banks are heterogeneous information systems; therefore it is difficult to build a common environment using existing distributed technologies. Web Services provide a means for different organizations to connect their applications with one another to conduct business across a network in a platform and language independent manner [1]. The paper will examine these benefits of Web Services and apply these features into the Internet banking transaction system regardless of platform, programming languages as well as existing backend system of banks.

1. Introduction

Nowadays, computer network has been playing its important role in business transactions. Many business firms have been using computer network to perform transactions with their partners. To perform these transactions, the network applications, which are called electronic-business applications, have been built for the purpose of carrying a business activity or collaboration. The electronic business activities are executed by network applications based on business process, which is considered as a precisely choreographed sequence of transactions that performs business tasks. These transactions are often operated automatically and, of courses, must follow workflow with conditions and rules that is established among business firms.

In banking activities, it is necessary to link banks together via computer network environment like above business process in order to improve their business activities. These applications usually have been based on distributed technologies such as EDI (Electronic Data Interchange) model, DCOM (Distributed Component Object Model) model, CORBA (Common Object Request Broker Architecture), etc. However, these existing technologies are facing with a several weaknesses. For example, most banks have heterogeneous sets of applications that do not adhere to one distributed technology. Although these applications can communicate each other with a distributed object interface, it is often costly and time consuming [10]. Moreover, above-mentioned models are less useful because the communication protocols do not work through firewalls and different banks may use different distributed object model.

Web Services technology is a new model for distributed

applications like above-mentioned banking transaction system. Web Services technology is based on a new set of standards such as XML (eXtensible Markup Language), SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language), UDDI (Universal Description, Discovery, and Integration), etc. With these standards, Web Services paradigm create a new approach for building data interchange applications which overcome above mentioned weaknesses of exist distributed technologies. This paper will examine the benefits of Web Services and present the authors' approach in applying Web services into the Internet banking transactions among bank systems.

2. The Internet banking transaction system

The Internet baking transaction system focused on this paper is the system that link bank systems each other so that transactions among banks can be operated by software applications automatically. Nowadays, most banks in over the world have applied Information Technology in their transactions. More and more consumers have been banking on the Internet. They access Internet to check their bank accounts, pay bills or transfer money [6] [7]. If they pay bills or transfer money to partner accounts in the same bank, the transactions will be processed inside the bank transaction system. Consumers only need to access on the bank web site to perform their transactions. In the case consumers would like to transact with the partner accounts in other banks, the transactions will be performed between bank software systems via computer networks. Transactions often have been implemented by network applications that allow data interchange via computer networks in order to perform these transactions with a few bank business rules automatically.

For communication to flourish, a coordinated system of signals that can be mutually understood is necessary [6]. That each bank has to build one channel (for data interchange) with each partner bank system makes the system heavy and in-uniform. Whenever one bank would like to join to the interchange banking system, the others have to change their implementation to adapt with their new partners. Vice versa, the new bank application has to implement all business rules to adapt with the interchange banking system. The following figure shows this model:

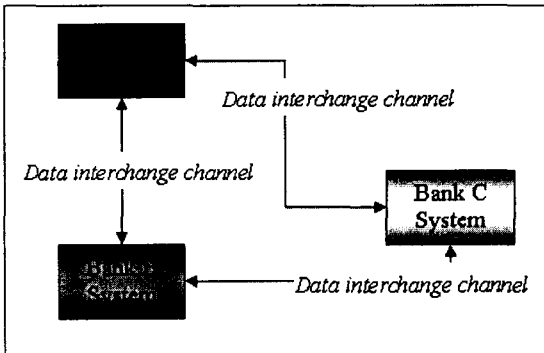


Figure 1 – Old model for banking transaction system

Each bank has its own system that is heterogeneous including software and hardware compared with the others. In order to communicate each other, the applications have to use distributed technologies but these technologies strongly coupled the endpoints and therefore could not become pervasive [2]. For instances, Unix RPC requires binary-compatible UNIX implementations at each endpoint, CORBA requires compatible ORBs, RMI requires Java at each endpoint, DCOM requires Windows at each endpoint. Compatibility and standards are the big problems for this heterogeneous interchange system. Moreover, it may be difficult or impossible to build applications that communicate via the Internet environment because most of these technologies can not work via firewalls.

Web Services paradigm is a new model which is based on XML and some related technologies. With Web Services technology, we can build programs to exchange data and business transactions in heterogeneous systems regardless of programming languages, operating systems, software vendors or hardware platforms. This technology is based on XML, SOAP message and HTTP protocol; therefore it can work via any firewall in the Internet environment. With these features of Web Services technology, this paper proposes a solution in interchange banking system which uses Web Services technology.

3. Web Services approach in the Internet banking transaction systems

Web Services technology is software components described via WSDL which are capable of being accessed via standard network protocols such as SOAP over HTTP [2]. Web Services are self-contained, modular business process applications that are based on the industry standard technologies of WSDL (to describe), UDDI (to advertise and

syndicate), and SOAP (to communicate). Web Services provide a means for different organizations to connect their applications with one another to conduct business across a network in a platform and language independent manner [1].

These features of Web Services technology can solve the weaknesses of other distributed technologies such as language and platform dependent, inflexible, disruption to existing interfaces of old systems. Using Web Services in banking transaction system, the banks only need to process the common XML structures (core of Web Services) for data interchange, the transaction will be implemented by any existing backend systems of bank without disruption of old system. XML is a standard that can be processed by any programming language because it has text file format. Web Services technology is based on XML; therefore, it is possible for any platform and programming language to build an application using Web Services. Moreover, in recent software industry, the standards related to Web Services technology are being widely adopted in the popular technologies such as Microsoft .NET Framework, IBM WebSphere, Sun One, etc [2][11][13]. Hence, Web Services can be applied into most existing bankend system. The following figure shows the model which this paper proposes

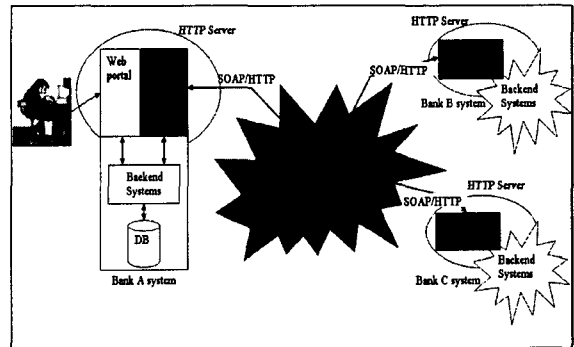


Figure 2 – Web Services approach in the Internet banking transaction system

3.1. Transactions between customers and bank

With these transactions, a web portal can be built so that customers who have a bank account can access and check their account information, and transfer money from their accounts to others one. In the case transactions which happen within the bank, the web application with any backend system can be built to implement. If the bank account transfers from the bank to others, the bank system must have an application to communicate with other bank system. In these transactions, the bank system only needs to implement an additional system that use Web Services to communicate with other banks in transferring money from one account in this bank to others in other banks. The paper assumes that transactions which perform inside one bank are implemented by existing backend system of the bank.

3.2. Transactions among bank systems

The aim of this paper is propose a solution for these transactions. As above mentioned, Web Services will be

applied in these transactions, between one bank system and the other in the Internet banking transaction system. Banks define together these transactions with some common XML structures and SOAP messages with banking rules. Each bank builds Web Services in its HTTP server based on these messages using exist backend system without disruption because XML and SOAP messages can be manipulated by any backend system (programming language and platform). The following figures show the workflow of this model

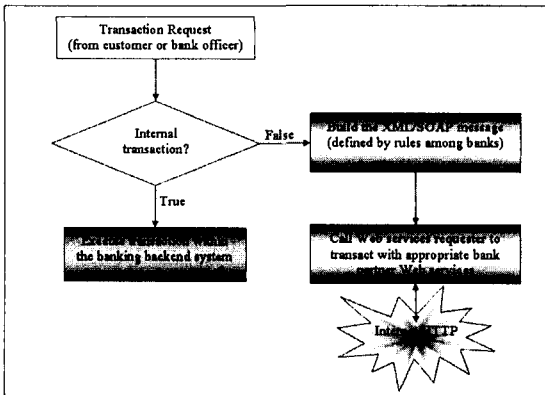


Figure 3- The workflow of a bank system when it receives the request of transaction in its system

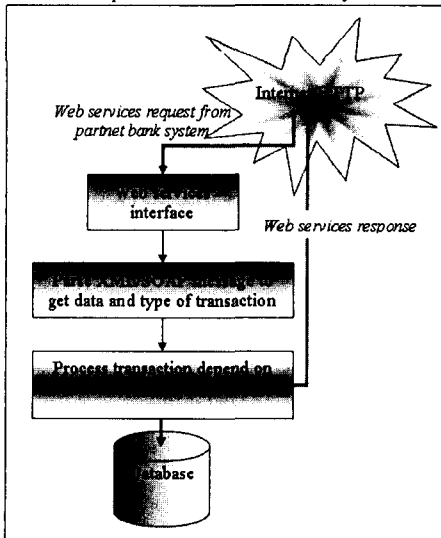


Figure 4 – The workflow of a bank system when it receives Web services request

When customers request for a transaction to other bank, the web application will request to Web Services requester. Web Services requester will build a SOAP message based on the defined structures. This message will be submitted to the appropriate bank system via HTTP/SOAP protocol. When the bank system receive a SOAP message, it will manipulate based on the defined banking rules, do the transaction and return the result for requester of partner bank system. These transactions are processed in same workflow and rules in any banks in the Internet banking transaction system. And it is the

same also for a new bank would like to joint to the system without worrying about its platform as well as backend system. The paper supposes that one bank has account in its each bank partner for account transferring.

Actually, the banking rules must follow banking transaction rules in the real world of each country. These rules belong to economic and banking major. Therefore, this paper has just only focused on technology aspect in order to build transactions. The above-mentioned transactions suppose that each bank has its own account in the partner bank, i.e. one bank is a customer of the other, using for transferring money of their customers accounts; therefore, transactions are executed directly between two bank systems.

4. Security for Web Services in banking transactions

Security is a very important problem in banking transaction systems. With banking transactions, the paper focuses on and divides to two kinds of transactions. The first one is transactions between customers and bank system. These transactions are based on HTTP protocol and between HTTP client and HTTP server. The second one is the transactions among bank system using Web Services. These transactions are exchanged between HTTP servers.

For communication between HTTP client and HTTP server, the paper use SSL (Secure Socket Layer). The SSL is protocol that allows a secure connection or information 'tunnel' between the web browser and a web server, based on a combination of public key cryptography and faster symmetric cryptography for encryption/decryption. This protocol provides confidentiality and integrity of data exchanged between the consumer browser and the merchant server [8]. Most HTTP clients and servers support SSL in secure transaction which is also called HTTPS. According to CommerceNet from ePSO-Forum (September 2001) and [9], SSL is the best solution for security between HTTP client and HTTP server.

Security for Web Services between HTTP servers takes more complicated. There are many technologies and protocols for Web Services security. The paper has been considered and chosen the standard called OASIS (Organization for the Advancement of Structured Information Standards) Web services security standard which was widely supported in application servers and development tools from several vendors. The standard defines extensions to SOAP envelop vocabulary including container for security tokens (Username, X.509 certificate, Kerberos ticket, XrML, XML Signature, SAML) and encryption details and guarantee for secure transaction in Web Services [4][5].

5. Implementation

In implementation of Web Services approach in the Internet banking transaction system, a transaction system with two banks has been built for testing. In one bank, Microsoft Internet Information Services (IIS) was used and web applications were implemented by C# programming language in Microsoft .NET Framework. Customers can use web browser to access to the bank system in order to check account information, transfer money to account in the same bank as well as the partner bank. The web application was

configured to SSL for secure transactions. In this backend system, a web services requester was adhered to existing system in order to process the transaction with the partner bank system. Web Services interface also was created to receive SOAP request from partner system and process the transactions based on banking rules which was defined in XML/SOAP messages.

In the other bank system, HTTP server was also IIS. Web Services interfaces were implemented by Visual Basic .NET. The system has worked well in communication between two different systems. Whenever transferring money request to the account of the partner system, the Web Services requester will build the SOAP message containing information of account transfer number, account receiver number, bank codes of two bank and digital signature for security. After built, the message will be sent to the partner banker system as Web Services request. When the partner bank system receives Web Services request, it parse data from SOAP message to get data and process the transaction based on the banking rules which were established. Confirm message will be sent back to Web Services requester of bank system after processing transaction success. This workflow has been implemented in both bank systems.

6. Conclusion and future works

Today, the Web Services paradigm promises to standardize the syntax and protocols used for communication between applications [10]. This paper presented an opinion of authors that apply benefits of Web Services into the banking system. The system in this paper is only simulation system; it is not tested in real world yet. However, it shows that Web Services can apply into banking transaction field with much improvement compared with different exist distributed technologies as above mentioned.

In the authors' opinion, to build the real Internet banking transaction system among banks, it is necessary to have the research of banking experts in order to establish banking rules among banks, including banks in one country or interchange among many countries.

In the future, we could investigate to do more research in a number of aspects related to Web Services technology such as SOA (Service Oriented Architecture), WS-Coordination, and WS-Transaction in order to help business streamline the integration process and implement more flexible business processes. Moreover, security is one of important challenges which is need spent more research in the Internet banking system.

References

- [1] Tom Freund and Tony Storey: Transactions in the world of Web Services, Rereach Paper, IBM (2002)
- [2] Mark Colan: A Technical Overview of Web Services, Presentation, IBM software Group *
- [3] Mark Colan: Using Web Services today (for tomorow), Presentation, IBM software Group *
- [4] Mark Colan: SOAP Security and Reliability, Issues and Solutions, Presentation, IBM *
- [5] Mark Colan: Making Web Services Secure, Presentation,

- IBM *
- [6] James J. McAndrews: Network Issues and Payment Systems, Federal Reserve Bank of Philadelphia BUSINESS REVIEW (1997)
- [7] Jackie Cuevas: The Internet Banking Horizon: Bleak or Bright for Community Banks? , Journal of Internet Banking and Commerce¹
- [8] Clara Centeno: Securing Internet Payments – The potential of Public Key Cryptography, Public Key Infrastructure and Digital Signatures, Background Paper No. 6, Institute for Prospective Technological Studies (2002)
- [9] JM Sahut & M. Galuszewska: Why does SSL dominate the e-payment market?, Journal of Internet Banking and Commerce²
- [10] Mark Hansen, Stuart Madnick, and Michael Siegel: Process Aggregation Using Web Services, Revised Papers of CAiSE 2002 International Workshop, WES 2002 (2002) 12-27
- [11] Mark Colan: SOA and Web Services: Where we are, where we're going, Presentation, IBM software Group (2004) *
- [12] Frank Leymann & Dieter Roller: Business processes in a Web Services world, Research Paper, IBM Software Group (2002)
- [13] Michael P. Papazoglou: The World of e-Business: Web-Services, Workflows, and Business Transactions, Revised Papers of CAiSE 2002 International Workshop, WES 2002 (2002) 153-173
- [14] James McAndrews and William Roberds: Payment Intermediation and the Origins of Banking, Journal of Economic Literature Classification Codes: E58, G21, G28 (1999)
- [15] James Snell: Automating business processes and transactions in Web Services, Research paper, IBM Emerging Technologies (2002)

* <http://www-106.ibm.com/developerworks/speakers/colan/>

¹ <http://www.arraydev.com/commerce/jibc/9811-14.htm>

² <http://www.arraydev.com/commerce/JIBC/0402-08.htm>