

Design of Financial Industry's Intermediary Business System based on Tuxedo

Xiaoyu Ma¹⁺ and Yunhong Guo²

¹Department of Computer Science And Engineering, Henan Institute Of Engineering, No 1, Zhongshan North Road, longhu conuty, Xinzheng, China

²Department of Electrical Engineering, Zhengzhou Railway Vocational & Technical College, No.2, Xingfu Road, Zhengzhou, Henan. China

Abstract. At present, each bank has its own intermediary business system, but most intermediary business systems are not satisfactory. In order to solve the problems in the intermediary business system, such as poor scalability and maintainability, and so on, this paper first analyzes the problems which intermediary business system of the financial industry faces, and then proposes the design of the intermediary business system of the financial industry based on Tuxedo, the design ideas of the intermediary business system proposed in this paper is relatively clear, has played a very good effect in the process of development and maintenance.

Keywords: Tuxedo; intermediary business; design.

1. Introduction

Intermediary business which is also known as agency business. Traditional intermediary business are: exchange business, business of credit letter, business of collection, business of colleague intercourse, behalf of the customer trading business, trust and leasing business. Agent party is also known as the third party. Intermediary business system (short for IBS): is application software system of processing intermediary business[1].

With the gradual open of financial market, the competition among the commercial banks will be fiercer and fiercer, the growth of the traditional business income is more and more difficult, launching the intermediary business will become important competitive advantage of commercial banks, the increment obtained through the intermediary business will be the banks' main income and one of competition focus[2].

2. Problems which the intermediary business system of financial industry faces

Currently, each bank has its own intermediary business system, but most business systems are not satisfactory, there are problems of the poor scalability and maintainability, and so on. Approximately analyzes, mainly embodies in the following aspects[3]:

2.1 Maintenance costs of the software system.

Because of the business expansion of the banks and the application of the new technology, banks often have the possibility of system upgrades, and the third party also upgrades system because of the new business demand. Intermediary business must strictly follow protocol of the banks and the third-party, the interface of either party changes, intermediary business must be modified accordingly, so, the maintenance is an important and long-term work. Usually software system upgrade of the banks and the third-party will be strictly followed the interface which is well made, but there is also the case that the interface changes, so the software system of

⁺ Corresponding author. *E-mail address:* hnzzmxy@yahoo.com.cn.

all external systems must be re-modified to adapt to the new banking financial hosts, many external software systems upgrade costs are too high, or can not complete upgrade in a limited period of time, have to use the new software system designed by software vendors, which causes a lot of waste.

2.2 The scalability of the software functions.

In order to improve the banking business and market expansion, banks need to support more third-party business or the new third party. Third-party business systems which are often developed by different software vendors have not an unified standard, and have much larger randomness when they define the interface protocol. However, the interface protocol defined by the third-party, for the intermediary business systems, must strictly follow the "standard". Therefore, the intermediary business must be very flexible to support the interface protocol of the third-party.

2.3 Consistency in distributed transactions.

A typical intermediary business transaction requires four trading parties which are network servers, intermediary business system, financial host system, third-party business system, and need to strictly guarantee that all transaction parties keep consistent for each business, that is , ensure the consistency of the transactions. For each business, four trading parties fail at the same time, or succeed at the same time, otherwise, will lead to the imbalance on the accounts.

The geographical distance of four trading parties involved in the transaction(as is shown in Fig. 1) is usually larger, and the current financial internal network situation is relatively poor and unstable in general, which is an objective status, maintaining the consistency of distributed transactions is a very difficult problem.

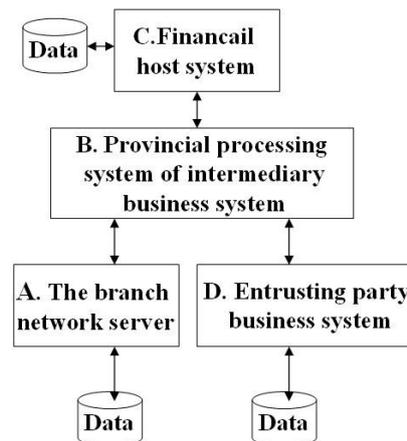


Fig. 1 The trading parties involved in the intermediary business

2.4 Fault-tolerant ability.

An objective environment of the intermediary business system which has been mentioned in the previous paper: the network is poor and unstable. Poor network conditions can directly lead to accounting transactions uncertain and inconsistent, resulting in the so-called "unilateral account", which is also a problem difficult to avoid, the intermediary business system must face these objective situation, add fault tolerance to ensure the consistency of the account service.

2.5 The ability to handle high concurrency.

At present, most of the intermediary business system use provincial focus mode, network servers are all over the province, each network server has a lot of terminals which handle business at the same time, and this puts forward very high demands for the ability that the system handles concurrent transactions. Once there is not enough concurrent processing ability, in the case of considerable pressure, the system's stability and reliability is facing a severe test. Therefore, the system must have a certain amount of throughput.

3. Design of financial industry's intermediary business system based on Tuxedo

3.1 Software architecture of the intermediary business system.

Intermediary business system is a large software system, its architecture is very complex, The involved area is extremely broad, includes software, hardware and networks, and so on[4]. This paper abandons the complicated business, focused on elaborating the software architecture of the intermediary business system, and how to take advantage of transaction middleware Tuxedo, constructs an intermediary business system, which well solved several difficult problems described in the previous paper.

The following outlined the software architecture of the intermediary business system from several perspectives of the network topology, system function deployment, hierarchy, and so on.

(1) The network topology of the intermediary business system

The network topology of the intermediary business system proposed in this paper is shown as Fig. 2:

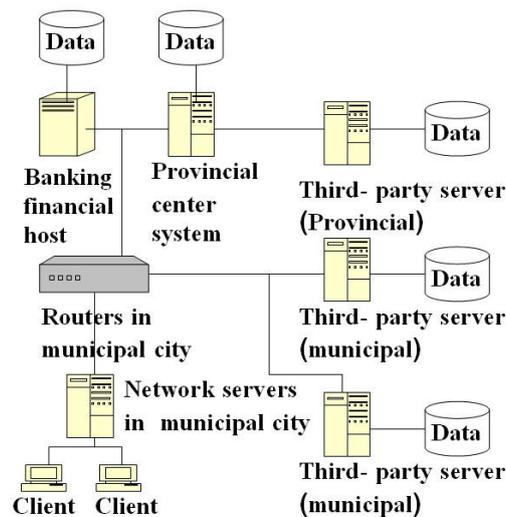


Fig. 2 The network topology of the intermediary business system

The provincial center systems of the intermediary business directly connects to the banking host. For some provinces, the unified third parties directly connects to the provincial center systems of the intermediary business. But for some of the municipal third parties, first connects to the routers of the municipal banks, then the routers transparently forwards transactions to the provincial center. There are their own network servers in all municipal cities, network servers do not handle specific business. Network servers complete the data exchange of the provincial centers through installing terminal server software, include the input and output, report printing, and other functions. Network terminals remotely login on the municipal network servers through telnet, run the terminal software, and complete all business operations on terminal software[5].

(2) System function deployment

The main function of the intermediary business application processing system is deployed in the provincial centers, including four subsystems: business processing system, accounting system, settlement system, management information system, two tool systems: configuration management tool system and loading operation tool system. The structure of business processing system is C / S, the provincial business processing system is responsible for centralized storage of business data across the province to complete the various types of intermediary business processing, support a variety of channels' access. Network front-end system, various types of pre-systems are as its clients which is responsible for initiating various types of transaction processing[6]. The client of provincial, municipal distribution business configuration management completes business configuration and part of the business management functions.

The structure of accounting system is C/S, B/S, the provincial accounting system is responsible for the centralized storage, accounting and processing of the whole province's accounting data. Provincial center, municipal city, county and bureau place the client software respectively. The accounting management clients of provincial bureau are responsible for the maintenance, query, parameter settings of the provincial accounting information, and the supervision and inspection of municipal city, county and bureau accounting work. The

accounting management terminals of the municipal bureau is responsible for downloading accounting information from the province's accounting system within scoping, supervision and inspection of city, county and bureau's accounting work, and so on. The accounting processing terminals of the city, county and bureau is responsible for the input and output of the accounting transactions, downloading and printing the accounting information, the real-time monitoring for network funds. the management of the important documents, manual input of the accounting transactions, payment, appropriation and the processing of bank's money on network directly deliver the business transactions to the accounting system of the provincial intermediary business for processing through the network front-end system.

The payment and settlement system uses C/S structure, the payment and settlement system of the provincial centers is mainly responsible for the deposit books and cards business which involve the savings accounts, the settlement of the cash business from allopatry. The operators in the province center is responsible for the completion of the daily processing and management in the client[7].

The structure of management information system is B/S, management information system of the provincial center is responsible for collecting and statistics analyzing all kinds of information, provide information query and statistics, data's upload and download and other functions for all levels of departments. managers can carry out all types of information inquiries, data entry and data download by web browser.

The provincial, prefectural and municipal distribution business configuration tools complete the business configuration and management.

The provincial center runs management tools to complete business's on-line loading, reconciliation and other business functions, In order to complete the prefectural and municipal reconciliation, can consider to place management terminals in prefectural and municipal end, but can only perform reconcile operations.

(3)The architecture of software system

Intermediary business system uses a hierarchical design, the low-level layer directly serves the high-level neighboring layer. From the hierarchical point of view, the software architecture of intermediary business system is shown as Fig. 3:

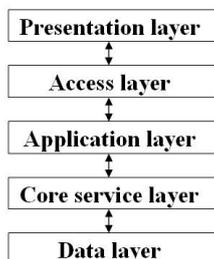


Fig. 3 The software architecture of intermediary business system

The software architecture of the intermediary business system can be divided into five layers, each layer is relatively independent. Thereinto, presentation layer is the interface of the human-computer interaction, there are common terminals, network terminals, ATM, POS and other forms. Access layer mainly completes the protocol conversion of each transaction end, and network's fault tolerance. Access layer abstracts the trading pattern and the communication protocol of each transaction end, defines a number of common protocol model, by increasing the configuration data in order to achieve the purpose of supporting the new protocol of the transaction end. Application layer completes the representation of the business logic, all the business logic can be described in the application layer through the business configuration and process configuration. The core service layer completes the analysis of the application layer's business logic rules, transaction control and security certification. Data layer completes the basic operation of the database.

3.2 The relationship of intermediary business system and Tuxedo middleware and the deployment of intermediary business system.

All the applications which are developed by using Tuxedo ATMI, their server-side program and client program are respectively compiled and connected by using the buildserver and buildclient, the server-side program is deployed on the Tuxedo Server, as a software component of Tuxedo, as the application server

container, Tuxedo maintains and manages the startup and stop of the application service program, and parameter configuration.

All application server programs deployed in Tuxedo are all the software component of Tuxedo. The process of development and deployment is as follows:

Step 1: Configure environment variables. On different operating systems, the approach of configuring the environment variable is very different, but at least concludes the following aspects: resources and its related lib position, Tuxedo and its related lib position, and so on.

Step 2: Coding, compiling, linking of Server and Client program. After the transaction program of C language completes coding, need to connect the correct library, and compile. Use buildserver can make the process easier. To use buildserver, must correctly set the environment variables, such as TuxDIR, PATH, LD_LIBRARY_PATH, etc.

Step 3: Configure the ubbconfig file, compile the configuration files into binary tuxconfig files, which are often called the ubbconfig files. Ubbconfig files can be regarded as the container which contains the application's startup information, after the ubbconfig is edited, we can use the tmloadcf command to compile the ubbconfig into the binary tuxconfig file, as the reference for startup.

Step 4: Starting systems and services. Tmboot command starts /T application of the Tuxedo system, creates the necessary IPC resources, starts the specified system in ubbconfig and the process of the application services.

Step 5: Adjustment of system parameters. Tests the application and adjusts the system parameters according to the environment, until the system performance is best. Only the reasonable configuration parameters can make the entire software system performance optimal.

3.3 The business logic configuration tools of the intermediary business system.

(1) Target

According to the intermediary business's characteristics, the intermediary business platform needs to have strong scalability and adaptability. Therefore, the intermediary business platform uses the parametric approach. We can achieve difference expresses of the same kind of business by adjusting the parameters, we also can add, delete and change the specific parameters of some kind of business. The management for parameters is achieved by the front-end configuration tools. The design of the configuration tool software should meet the requirement of the parametric intermediary business platform, which can flexibly and easily complete the parametric work.

Configuration tool belongs to the subsystem of the intermediary business software system, its main task is to create, modify, and delete the configuration files which the run of the intermediary business needs. Configuration tool should have the function of the users' management, have authentication and permission management for the users who use this system.

(2) Principle

The software uses object-oriented language, and should have a clear structure and normative definition.

(3) Data security

In order to avoid on-line business faults caused by the update of parameters, tables in the frontal parameter database use dual-table mode, that is, create two sets of the parameter tables which have the same structure in the different table space, one for business operations (RunTable), The other for the configuration of the tool parameters (SetupTable). After the new updated parameters have been tested, then update RunTable.

Configuration tool only operates on SetupTable, establish a dedicated user for the operations on SetupTable. The user's permission is limited to SetupTable, no RunTable permission. Taking the human's error operations into account, have data backup before modifying the parameters, when finish modifying, the system should prompt the modified content, if confirm that it is a correct operation, then finish it, otherwise, recovery the information of original parameters. In order to prevent errors, there should be a synchronization control mechanism between RunTable and SetupTable to ensure the version of the RunTable and SetupTable is consistent.

(4) Data conflict

Public information should use the principle of the central unified configuration management, municipal business information is configured by the municipal city. To prevent abnormal phenomena, such as data duplication which is caused by manipulating the same data by multiple users simultaneously, the parameter updates use the transaction commit mechanism[8].

4. Summary

This paper first analyzes the problems which the intermediary business system of the financial industry faces, then proposes the design of the financial industry's intermediary business system based on Tuxedo, including: software architecture of the intermediary business system, the relationship of intermediary business system and Tuxedo middleware and the deployment of intermediary business system and the business logic configuration tools of the intermediary business system. The hierarchical design ideas of the intermediary business system proposed in this paper is relatively clear, has played a very good effect in the process of development and maintenance.

5. References

- [1] V Basili, L Briand, W Thomas. Domain Analysis for the Reuse of Software Development Experiences[C]. In Proc. Of the 19th Annual Software Engineering WorkshoP, NASA/GSFC, Greenbelt, MD, Aug, 2010
- [2] A Campbell, G Coulson, M Kounavis. Managing Complexity:Middleware ExPlained[J]. IT Professional, IEEE Computer Society, May, 2010.
- [3] M D Lubars. Wide2spectrum Support for software Reusability[M]. In workshop on Software Reusability and Maintainability. National Institute of Software Quality and Productivity, Feb, 2010.
- [4] Garango, R Prieto2Diaz. Domain Analysis: Concepts and Research Directions[A]. In R Prieto2Diaz, GARango1Domain Analysis:Acquisition Of Reusable Information for Software Construction[M].IEEE Computer Society Press, Jul, 2010.
- [5] P Bernstein. Middleware:A Model for Distributed System Services[J].Communications of the ACM, Sep, 2010.
- [6] Matthew J Moyer, Mustaque Ahamad.Generalized Role2 based Access Control[C]. Proc.of the 2010 Intl. Conf. on Distributed Computing Systems(ICDCS), Oct, 2010.
- [7] David F Ferraiolo, Ravi Sandhu, et al. Proposed NIST Standard for Role-based Access Control[J]. ACM Transactions on Information and Systems Security, Apr, 2001.
- [8] Ravi Sandhu, Edward J Coyne. Role2 based Access Control Models[J], IEEE Computer, Sep, 2010.