Journal of Applied Business Research *Volume 20, Number 4*

A Unified Accounting Information Framework To Modeling Bank Accounting Systems

Fengyi Lin, (Email: linfengyi@mail.chihlee.edu.tw), Chihlee Institute of Technology, Taiwan, R.O.C.

Abstract

This paper discusses the use of distributed middlewares as essential tools for facilitating electronic exchange of standard business document between managers, financial institutes, and trading partners in the banking sector. Internally, companies can benefit by creating information architectures that allow systems to easily exchange data. One less expensive and disruptive option that applies to most banks/financial institutes is used traditional mainframe (legacy) system with an array of distributed middlewares to overcome the aforementioned limitations. This paper focuses on developing a new distributed processing architecture based on client-server technology called UAIF – Unified Accounting Information Framework. UAIF is designed to assist managers/financial institutes with a transparent access to information anywhere on the LAN or WAN from any desktop and to meet management specific needs so that some of the accounting and financial works can be widely used for World Wide Web (WWW) applications via Internet or Intranet.

For concept verification, we utilize UAIF to modeling a bank accounting system, which is based on an industrial standard CORBA architecture, XML and OMG General Ledger Facility. This methodology integrates enterprise accounting information system (AIS) with distributed systems via Internet, Intranet, and Electronic Commerce.

I. Introduction

The banking sector is undergoing significant changes in its accounting information system strategies, because of the challenges posed by globally competitive market and ever-increasing demands on information technology (AICPA 1998；Liang et al., 2001). The world's major banks spent billions of dollars building information and transactional services for the Web over the past several years. Recently, corporate customers are increasingly interested in online offerings that present them with a fundamentally better way of doing business. Internet banking has made the most headway and is reaching critical mass among corporate and institutional customers in financial or accounting related information such as cash management, fixed-income trading, foreign exchange trading, payment and letters of credit (Rombel, A, 2003). In addition, quality, security and risk assessment are vital characteristics of the sector and of its information systems (Ramani and Pavri, 1994 and Dutta and Doz., 1995). Fully integrated software packages -- SAP, PeopleSoft, Baan, and Oracle provide many capabilities such as entry and analysis of transactions in real-time, Internet and Intranet connectivity, enterprise-wide budgeting, multidimensional analysis, and workflow (Lucy 1999). However, the above IT investments are costly and time consuming, AICPA 1998. Moreover, they could disrupt the existing systems and organizational structures.

Current accounting information systems (AIS) seek to link servers over the Internet to exchange data which are subject to the following considerations. First, today's computing environments require a high degree of interoperability among heterogeneous computer systems and applications (Orfali et al., 1995；SSI Ltd. et al., 1998). However, most organizations currently employ closed mainframe systems; each department or external trading

partner can build its application with its own proprietary operating systems, application programs, and data files. Namely, the financial institutes which gather information for investors need to face the inter-application problems between different departments and trading partners (Petrie, 1996；Lin et al., 2000). Second, bank AISs are inter-connected within the heterogeneous network environment of LANs and WANs (such as Ethernet, FDDI, and ATM) where the different protocols and characteristics restrict the reusability, portability and interoperability of their accounting information software (Clark, 1994, Wallnau and Rice, 1995). This has led to an increasing emphasis on a distributed processing architecture based on client-server technology (Lucy, 1997). Third, many organizations including bank sectors are now recognizing the importance of early warning system controls such as embedded modules or real-time detection modules which provide an integrated diagnostic view of on-line, real-time system. These modules contain a set of manager defined rules. Exceptions to these rules will trigger alarms that are intended to call the manager's attention to any deterioration or anomalies in the systems. However, the above IT technologies are suggested to implement as early as possible in the legacy system development life cycle since a heuristic research indicates that after the design phase, it will cost 4 times as much to retrofit controls into a system; and after implementation, it will cost 16 times as much to retrofit controls into a system.

We therefore propose an open system oriented approach - Unified Accounting Information Framework (UAIF) that is based on an industrial standard CORBA architecture to overcome the aforementioned limitation. UAIF's mission is to promote true distributed client-server processing and inter-applications communication in order to facilitate electronic exchange of standard business documents between third parties. Instead of scrapping the legacy systems, this framework takes advantage of distributed object technology, on-line detection system (Booch 1991, Clark 1994, CORBA 1995), XML and OMG General Ledger (GL) Facility, which can improve the quality and accuracy of accounting information for specific industry and can be viewed as an enabling technology for continuous reviewing and auditing (AICPA, 1995, Booch, 1991).

The UAIF can play an effective role in modern organization only if its IT architecture recognizes the following developments:

1. UAIF is not an isolated and discrete organizational project but a technological innovation that is continuous over time. With a vast majority of core platform application functionality completed already in the form of vendor developed components, future unified AIS should consider to use common, industry-wide domain interfaces.
2. To make information useful for decision-making, accounting information must be reported on a timely basis. UAIF will speed up and reduce the expense of collecting and processing data and will facilitate real-time measurements. For example, financial institutes can collect accounting information within seconds in order to more accurately predict the company's performance.
3. Under the UAIF environment, "workflow" software often is used to control electronically transaction creation, flow, authorization and storage. Not only managers but also financial institutes can review transactions at any stage in the process. UAIF adopts a transaction-based approach that involves continuous transaction monitoring and a series of short compliance reviews for individual branches. Furthermore, UAIF's monitoring can avoid interruptions to normal workflow and regular business process.

II. Unified Accounting Information Framework

UAIF consists of three important emerging technologies: Distributed object technology, on-line detection operation and XML. The framework is depicted in Figure 1.

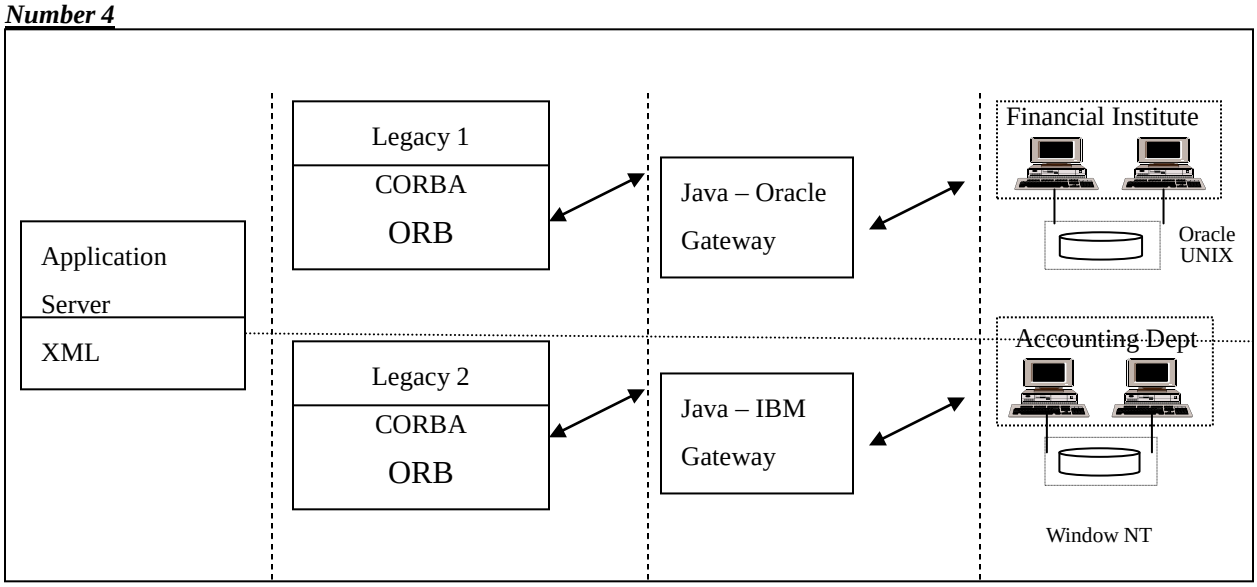

Figure 1. Unified Accounting Information Framework

Distributed Object Technology

The next wave of client/server technology must be able to handle multiuse, consumer-to-business, and business-to-business transactions across multiple servers distributed over global networks. Existing distributed system such as the Object Management Group's CORBA, the Open Group's DCE, Sun Microsoft's Enterprise JavaBeans, or Microsoft's Distributed Component Object Model (DCOM), not only permit applications and data to reside on different platforms such as UNIX or NT system, but also require them to work together seamlessly and support various business processes. In all of these environments, server applications are designed to advertise interfaces to services, or functions, that they offer. Clients locate these services by searching a directory and bind to the desired services using the information stored in the directory.

Common Object Request Broker Architecture (CORBA) is one of the well-known distributed middleware that is proposed by a consortium – Object Management Group (OMG). CORBA promotes interoperability, extensibility, portability, distributed processing and asynchronous operations all within an environment adhering to open systems standards and providing improved performances (Lee 1995). If widely adopted, CORBA may have a profound impact on AIS software marketplace.

On-Line Detection System

As the communication technology evolves, it will become possible to run on-line detection applications in distributed environment. By using UAIF, the real-time accounting system can perform trading service with the middlewares supports such as ODP, CORBA or OSF/DCE. On-line detection system is able to design for reviewing and analyzing data flows through the system continuously (e.g., transactions in inactive accounts, write-offs of accounts receivable or inventory, deviations from predetermined purchase and sale policies, and the frequency of certain types of transactions) using a set of manager defined rules. The plug and play (PnP) mechanism can set embedded modules into the legacy system through CORBA facility more flexible without changing the original AIS systems. Bank manager amounts to an analytical review technique since constantly analyzing a system allows the manager to improve the focus and scope of the operation. Internal control checkpoints can easily shift control monitoring as previously programmed and the embedded model can be set up for red flags at certain financial limited. UAIF improves the quality and timeliness of corporate controls and consequently increases the efficiency of

financial and accounting productivity. The framework of UAIF is depicted in Figure 1.

The OMG (Object Management Group) General Ledger Facility submits proposed interfaces to enable interoperability between General Ledger systems and accounting application. The GL interfaces comprise a framework (in the object-oriented sense) that supports the implementation of accounting client applications, for example: accounts payable, accounts receivable, payroll, and so forth. This is the area for tremendous growth in the e-business model and a critical place to develop GL facility to help integrating AIS heterogeneous environment. Figure 2 gives a high level description of the General Ledger Facility interfaces.

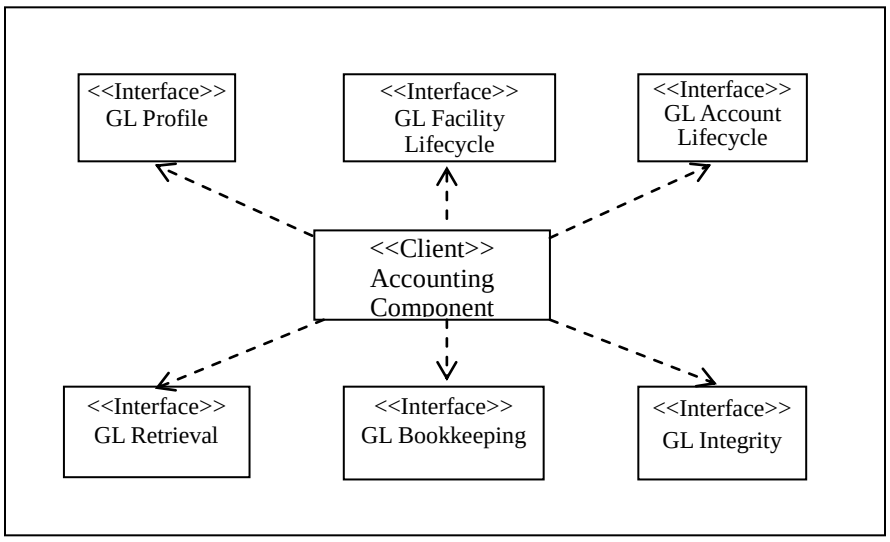

Figure 2 General Ledger Facility

XML vs. CORBA

Internally, companies can benefit by creating information architectures that allow systems to easily exchange data. XML stands for eXtensible Markup Language. It's a simplified subset of a previous markup language standard and was devised by a committee of the World Wide Web consortium. UAIF depends on Internet protocols or standards to markup the data and to make movement of data from system to system possible. Consider a company with several different general ledgers running in various divisions around the world. Web services could be created where information in each separate general ledger, coded in XML, is retrieved upon command from each system. Accordingly, accounting department can alleviate the dependencies on proprietary vendor's implementations of accounting applications, both hardware and software, and promote inter-applications communications. CORBA creates sophisticated, distributed object systems on heterogeneous platforms. It allows users to connect disparate systems and form object architectures. Another advantage is accounting department might be able to upgrade its software at different period of time with different vendors since the programmer can use CORBA support language to write applications without knowing what language the implementation uses. The ORB will handle all client requests and direct it to the right object, invoke right operation then take the data back to the client. For instance, as shown in Figure 1, accounting department can adopt UNIX system, whereas outside financial institutes might utilize NT system. As a result, once the configuration of the server (implementation), a distributed, heterogeneous environment, is setup; all the clients can share it.

XML combined with CORBA allows information to connect freely to other applications and; then read messages sent from them. XML is intended for the storage and manipulation of the making up humane-readable documents like Web pages, while CORBA ties together cooperating computer applications exchanging transient data that will probably never be directly read by anyone. Neither of these technologies will replace the other, but instead they will increasingly be used together. For example, a message transported to a remote server using CORBA could contain a request that the computer send data back from the remote server, such as the operations budget detail for prior month. The remote system will respond and fulfill the information request.

With UAIF, greater access to company data for shareholders and financial analysts become possible through computer networking. UAIF combined with XML, CORBA and OMG GL Facility technologies is a framework that provides the potential for connecting different institutions into one large system without changing their legacy systems.

III. An Application Design Example – Bank Accounting System

Through utilizing the General Ledger Facility above, we could start to design a bank accounting information system under an UAIF environment. The steps and procedures of OMG GL Facility employed by the bank institution should be used to ensure a controlled development and implementation. This system has four external entities: Branch; Management; Outside financial institution and Customers. Events are identified for each external entity. An event is a starting point for trigging the systems.

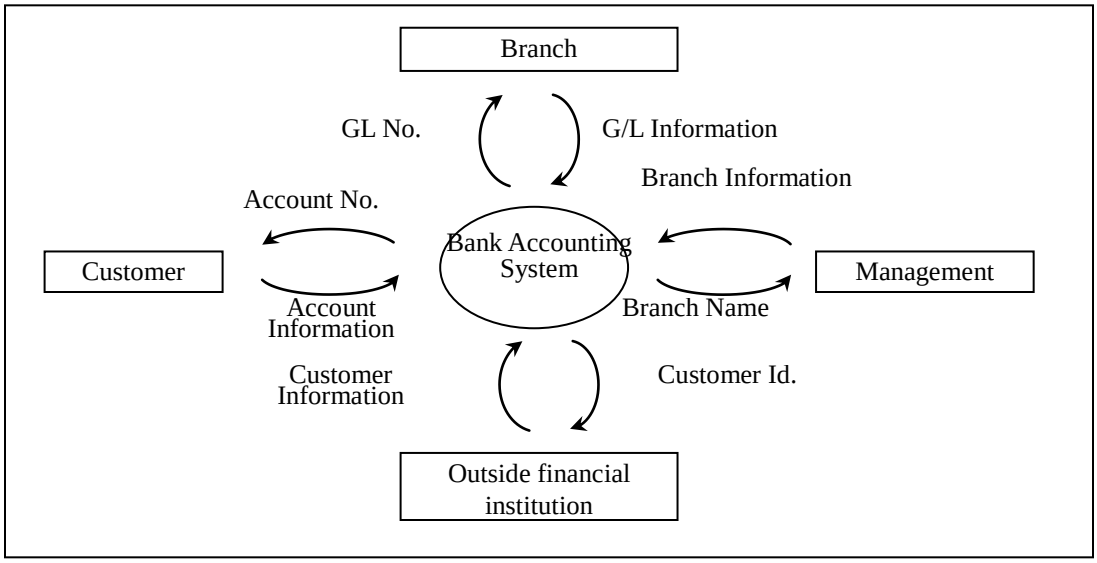

Figure 3 System Scope Diagram

Once the potential areas of AIS scope and events have been identified, the next critical step is to design proposed standard interfaces and their semantics based on GL Facility. Without these common GL interfaces defined, an exponential number of accounting interfaces may result in integrating AIS between the bank institution, clients and stakeholders technically unfeasible. Four basic bank object classes (Customer, Account, Branch, and Transaction) modeled for this bank institution as shown in Table 1.

A basic attribute is generated from a single object class. Each class contains association relationship with some other associators. For instance, the basic class, Customer as depict in Table 1 has association relationship with Account (N) and Transaction (N). We use customer ID to identify different customers and each customer objects include the same attributes. Every client invocation uses the Interface Definition Language or IDL to specify all interfaces and supporting data types. The IDL enables an object written in different languages and/or implemented on different platforms to link to other remote objects. For example, if one of the bank branches used a large class written in C running on a UNIX machine, a developer can use CORBA to permit that routine to communicate with another branch system written in Visual Basic, C++ or even Windows NT environment. In Table 1, customer class, for instances, contains 8 attributes that can be further designed as CORBA interfaces as follows:

Table 1 Accounting Interface

Class Customer	Class Branch	Class Account	Class Transaction
Attributes Cust_id, Cust_SSN Cust_Name, Cust_Job Cust_Phone, Cust_Addr Cust_Credot, Cust_Asset	Attributes Bran_Code Bran_Name Bran_Addr Bran_Phone	Attributes GL_Acct_Ref, GL_name GL_Acct_Balance GL_Acct_MonthlyBalance GL_Acct_YtdBalance	Attributes Trans_Code Trans_Date Trans_Type Trans_Amount
Type:	Type:	Type	Type
Customer ID	Branch #	Cash, A/R, A/P, Equity, Revenue, Expense	Deposit, Loan, Credit card
Associators:	Associators:	Associators:	Associators:
Account (N) Transaction (N)	Account (N) Customer (N) Transaction (N)	Branch (1) Customer (1) Transaction (N)	Account (1) Branch (1)

One of the most challenging steps considered in the development of UAIF is to identify high-risk areas and potential fraudulent activity. It can effectively perform continuous operating financial information and monitoring systems. For example, managers can build their own management module such as deposit module, loan confirmation module, due credit amount module and so on to monitor internal control of important daily transaction through PnP function. The appropriate members of management could flag all high-risk areas for review to ensure that documents have been properly prepared, reviewed, and approval.

Many organizations publish detailed manuals about how to budget and give their managers spreadsheet for consistent input to accounting. The manuals are time-consuming and, unfortunately, often ignored. The spreadsheets are appropriate but don't provide detailed budgeting online, nor do they allow for numerous rollups under varying conditions. UAIF can search and provide benchmarks for prior-year data and often explain those numbers themselves. For example, strong cash management offerings allow a treasurer to check a balance to see if a payment is missing, initiate the payment and get a confirmation in one or two clicks on one screen. And with customization corporate users can tailor their cash management-reporting template to be on alert for certain balance thresholds and execute transactions automatically. UAIF offers customization and business-process-changing value because it can provide integrated platforms, linking Web applications with one another and with other bank channels, legacy systems and internal databases.

IV. The Advantage Of UAIF

From a control and audit standpoint, the characteristics of unified accounting information environment facilitate the addition or enhancement of preventive and detective controls. For instance, with standard objects via homogeneous object bus over Internet and intelligent workflow software, bank managers can review branch's General Ledger system at any stage in the process through proper authorization. Another compelling reason for the

use of UAIF is this framework provides the potential for connecting different organizations into one large system without changing its original legacy system. It provides a robust platform for significantly enhanced assessment, extraction, and query tools for managers and other users.

Under this bank accounting information system framework, additional checkpoints and review procedures effectively shift monitoring of controls back to management. Manager can set a manager module with a Plug-and-Play function to on-line detects exception and potential frauds with distributed object and high-speed networks. One example is the payroll module providing taxation magnetic reporting capabilities to meet government requirement. Government can take advantage of this system by using CORBA interface to access a company's original transaction information.

Bank information system can support substantive transaction test by using random, stratified, and judgmental practices. For instances where high dollar loans, deposits, or withdrawals over a certain amount, or every 60^{th} invoice for amounts up to US $200,000 can be delivered electronically to the internal audit departments or to managers for review prior to payment. It also enables the managers to track the profitability of each mortgage contract or individual customer's credit so that new contracts can be adjusted properly when they are up for renewal.

A paradigm shift occurs when UAIF enables managers and financial institutes to use a transaction-based approach that involves continuous transaction test and a series of short compliance reviews for individual departments. UAIF improves investor and analyst's ability by accessing into a company's financial information, thereby lowing their uncertainty over perceived risks of investing and providing them with credible, reliable information.

V. Summary And Discussion

This paper introduces a unified accounting information framework where we advocate that some of the accounting managers/financial institutes work can be conducted electronically over Internet with support of information technologies. This technology can be used for greater disclosure and may have a profound impact on corporate governance. We have identified three emerging information technologies include distributed object technology, on-line detection, XML and OMG General Ledger Facility to constitute a distributed framework.

For the purpose of concept verification, we present a prototype bank accounting systems based on CORBA. The successful implementation of distributed object systems will enable managers/financial institutes with an increased level of comfort that transactions continue to be processed in an accurate, complete and highly controlled environment regardless of where they are located or who has designed the application.

This UAIF depends on distributed middleware standards, i.e., CORBA, DCOM, or Java RMI, to enable the interconnections of customer, branch, managers, and the rest of WWW. All these middleware technologies are at their infantry stage; this implies that they are evolving standards. We expect an explosive application of these technologies may take place when they become more mature.

Acknowledgments

This research is supported in part by the National Science Foundation under grant NSC92-2416-h-263-001.

References

1. American Institute of Certified Public Accountant, AICPA.," The IT committees' top 10 list," *Journal of Accountancy*, 183 (2), 1998.
2. American Institute of Certified Public Accountant, AICPA,"Auditing in Common Computer Environments", *AICPA*, New York, and NY, 1995.

3. Birtcher, J.," Auditing client/server information processing", *EDPACS* 22 (8): 1-4, 1995.
4. Booch, G., "*Object-oriented Design with Applications"*, The Benjamin Cumings Publishing Company, Inc., 1991.
5. Cash, J. I., A. D. Bailey, Jr. and A. B., Winston, "A survey of techniques for auditing EDP-based accounting information systems", *Accounting Review*, 3 (4): 813-832, 1997.
6. Clark, B., "Solomon III", *Management Accounting* 75(7): 66-67, 1994.
7. "The Common Object Request Broker Architecture and Specification" (*CORBA*) Revision 2.0. http://www.omg.org/corba/corbiiop.htm., 1995.
8. Cushing, B. E. and M. B. Romney, *Accounting Information Systems-A Comprehensive Approach*, 5th ed. Addison-Wesley, 1990.
9. The Component Object Model Specificati.COM Available at http://www.microsoft.com/oledev/olecom/title/htm.
10. Donahue, M., "Entering a new era in is audit and control", *EDPACS* 21 (12): 1-5, 1994.
11. Dutta, S. and Doz, Y., "Linking information technology to business strategy at Banco Commercial Portugues," *Journal of Strategic Information systems*, 4, 89-110, 1995.
12. Garceau, L. R., J.D. Schoeffler, and A. Fazio. "Middleware security: CORBA or DCOM?" *IS Audit & Control Journal*, Volume IV 4: 34-41, 1999.
13. Lee, J. E., "The Dispatcher: a new architecture for distributed client-server GIS processing", Ninth Annual Symposium on Geographic Information Systems in Natural Resources Management. *GIS World*. Part 1: 170-3, 1995.
14. Liang, D., F. Lin, and S. Wu., "Electronically auditing EDP systems with the support of emerging information technologies," *International Journal of Accounting Information Systems* 2: 130~147, 2001.
15. Lin, F., D. Liang, S. Wu, and R. M. Yang, "An integrated auditing architecture for Internet and information system design under a CORBA environment", *Review of Accounting Information Systems* 4 (1), 2000.
16. Lucy, R. F., "Electronic document management: an internal control paradigm shift", *IS Audit & Control* 5: 31-35, 1997.
17. Lucy, R. F. "Is auditing: the state of the profession going into the 21st century", *IS Audit &Control Journal*, 4: 44-49, 1999.
18. Mohrweise, L. C., "Usage of concurrent EDP audit tools", The EDP Auditor 3: 49-54, 1988.
19. Orfali, R., D. Harkey, and J. Edwards,"Client/server components: CORBA meets open-door", Object Magazine 5(2): 55-9, 1995.
20. Petrie, C., "Agent-Based Engineering, the Web, and Intelligence", *IEEE Expert*, 1996.
21. Petrie, C., "Java Remote Method Invocation Specification". Revision 1.42, JDK 1.2, Javasoft Ltd., Mountain View, CA USA, 1997.
22. Ramani, K.v. and Pavri, F., "IT supports business strategy growth at the Development Bank of Singapore", *Journal of Strategic Information systems*, 3, 327-337, 1994.
23. Rombel, A., "Next step for Internet banking", *Global Finance*, Vol. 17, Issue 2, pp. 43-44, New York, February 2003.
24. SSI Ltd., et al. General Ledger Facility, OMG DTC Document finance/98-07-02, 1998.
25. Wallnau, K.C., and J. Rice., "ORBs in the midst: studying a new species of CASE integration mechanism", Seventh International Workshop on Computer-Aided Software Engineering. *IEEE Compute*. Soc. Press: 49-58, 1995.
26. Yi, S.Y., J. Lee. and H. Shin. "Operating system support for the trader in distributed real-time environments", Proceedings of the Fifth IEEE Computer Society Workshop on Future Trends of Distributed Computing Systems, *IEEE Compute*. Soc. Press: 194-9, 1995.
27. Zoladz, C., D. A. Watne, and P. B. Turney, "Auditing in an Integrated EDI Environment", *IS Audit & Control*, Auditing EDP Systems, and 2nd ed, 1990.