Applying Mobile Agent to a Mobile Stock Intermediary Services System Development

Wen-Hsiung Wu[1], Po-Chang Ko[1], *Ping-Chen Lin[2], Ming-Hua Su[1]
[1] Dept. of Information Management, [2] Institute of Finance and Information
National Kaohsiung University of Applied Sciences
{whwu,cobol,lety}@cc.kuas.edu.tw

Abstract

Due to the radical changing of the global economy, a more precise stock valuation helps providing important judgment principles to decision-makers and investors. With the advent of the third-generation (3G) or future forth-generation (4G) Internet, the mobile commerce (M-Commerce) will become increasingly important. In addition, the mobile stock investment decision support system attracts great interests for professionals, such as stockholders, bondholders, financial analysts, governmental officials, and even the general public, recently. This study introduces a Mobile Agent-based Stock Intermediary Services System (MASISS) framework based on the mobile agent perspective to provide ubiquitous and seamless transaction activities for financial institutions. It also helps customers to make a more precise decision in the current intense commercial competition environment. For building distributed enterprise systems, The MASISS framework is developed in an integration of J2ME and J2EE environment with cross-platform portability, a huge server-side and client-side deployment base, and coverage for most W3C standards..

Keywords: *Mobile agent, Mobile stock intermediary services, Stock investment decision, Wireless handheld devices.*

1 Introduction

Investment decision making plays an increasing important role in the knowledge economy age, because the global economy has radically changed. It helps many professionals, such as stockholders, bondholders, financial analysts, governmental officials, and even the general public, to make a more precise decision in the current intense commercial competition environment. However, in the face of a rapidly fluctuating market and a vast amount of information to be digested, the information asymmetry makes it hard for investors to make precise and real-time decisions. Now that the Internet has become ubiquitous, most securities companies and investment organizations provide their own e-stock intermediary service systems for the benefit of investors. These systems allow investors to obtain real-time information, conveniently place orders over the Internet and thereby provide economic benefit. Even the widespread use of wireless handheld devices (including mobile phones and PDAs) has gradually moved mobile stock investment into the limelight in recent years. The use of mobile services not subject to restrictions of time and space is better able to meet investors' needs. As a consequence, as far as securities companies and investors are concerned, the use of mobile stock intermediary services to provide stock information and implement transactions has become a necessity.

**Corresponding author: Tel.: +886-7-3814526 ext.7508. E-mail address: lety@cc.kuas.edu.tw (Ping-Chen Lin).*

From the point of view of the market, some companies – such as E-Ten Information System Company – are trying to provide mobile stock intermediary services. Nevertheless, its main functions provided by these services are identical with those of e-stock intermediary service systems, which include market trends, individual stock trends, category rankings, and technical analysis. While these functions offer the advantages of real-time information, real-time order placement, and mobility, they make no attempt to provide advanced individualized services via agent-based intermediary services. As far as securities companies are concerned, the establishment of a mobile agent-based intermediary service to serve investors can not only disperse transaction volume and reduce bandwidth use, but also provide the advantages of cross-platform compatibility, autonomy, and interdependence (Picco, 2001; Manvi and Venkatardm, 2004).

The characteristics of agents have recently started to attract great attention in financial fields. For instance, Chen and Liao (2005) used an agent-based stock market model to analyze the relationship between stock volume and price. Nevertheless, fewer researches have applied the features of mobile agents to the mobile stock investment decision-making environment. Furthermore, although most existing commercial mobile stock intermediary services provide at least prototype functions, they still fail to actively provide individualized services. For example, an investor may only be concerned about the associations between the rising prices of stocks in a certain category. Mobile stock intermediary services must therefore take investors' specific needs into account.

Based on the above considerations, this study proposes a framework for a "mobile agent-based stock intermediary services system" (MASISS). This study also develops MASISS in a JEME and J2EE environment with cross-platform portability to build a huge server-side and client-side deployment based enterprise system. We also assess MASISS by using the example of association between stocks in a single category. MASISS not only can provide a reference for securities companies developing mobile stock service functions but also provide individualized assistance to investors making mobile stock investment decisions.

2 Intelligent Mobile Agents with Mobile Commerce Applications

Agents are autonomous objects created for dynamic and distributed applications that are responsible for executing designated tasks (Woodridge and Jennings, 1995). Use of agents and knowledge can improve transactions between information suppliers and customers, but place a heavy workload on the intermediary and thus limit the number of participants that can be served by the intermediary. In contrast, mobile agents can solve the heavy workload problem, and can be transported on different systems after being executed asynchronously and autonomously, carrying with them their program code, current state of execution, and any data obtained. Furthermore, mobile agents can communicate with one another (Chess et al., 1995; Picco, 2001).

The main characteristics of mobile agents include: (1) Remote distributed processing: The distributed processing program to be executed is sent to other computers for execution. The program code is compressed before transmission, communications volume is minimized, and the burden on computer and network is distributed. (2) Asynchronous processing: When the transmitting server sends the program to a receiving computer, the two ends do not need to stay in continuous communication; mobile agents can be executed independently on a receiving computer, which sharply reduces network traffic. (3) Simplification of distributed program design: As long as a run time system is installed on every computer, all computers will possess a mobile and autonomous processing ability. As a result, mobile agents possess the advantages of disconnected operation, less network traffic, roaming ability in a heterogeneous environment, support for electronic commerce, ease of development, personalization and high flexibility, and real time application (Lange and Oshima, 1999; Picco, 2001; Cao et al., 2004; Manvi and Venkatardm, 2004).

Because mobile agents possess the foregoing characteristics and advantages, they are being used by researchers in many fields for an increasingly wide range of applications. Mobile agents were formerly used mainly for network management, monitoring, and scheduling, including in heterogeneous network management and integration systems (Du et al., 2003) and dynamic scheduling systems (Hiroyuki et al., 1997). In e-commerce environments, mobile agents have been applied to e-trading, brokerage, auction, and e-marketplace functions, including the developments of MAGENT, BROKERAGE, and Nomad systems (Dasgupta et al., 1999; Jung and Jo, 2000; Sandholm and Huai, 2000; Du et al., 2005). We have found that a small number of researchers have applied mobile agents in the applications of financial decision-making. Specifically, Chen and Liao (2005) proposed an agent-based stock market model, and investigated linkage between stock returns and trading volume.

Mobile commerce has become an important trend in recent years. Quah and Lim (2002) suggested that we are carrying out research into the use of mobile agents in wireless handheld devices because they will overcome wireless network limitations such as small bandwidth and intermittent disconnection problems. As mentioned in the previous section, agents possess autonomy and several other advantages. We have found that the mobile agent concept has been applied to web search, tourist guide, and auction applications in relevant studies, and is developed as the Search Sweep, Gulliver's Genie, and MoRAAS systems (Zerfiridis and Karatza, 2004; O'Grady et al., 2005; Shih et al., 2005). This shows that mobile agents can be readily applied to mobile commerce issues.

In summary, we have found that few studies have applied the features of mobile agents to mobile stock investment decision-making. Therefore, we try to develop a mobile agent-based intermediary service system for both securities companies and mobile investors in this study.

3 Framework of MASISS

This study proposes a "mobile agent-based stock intermediary services system" (MASISS) based on the concept and features of mobile agents, and taking the mobile stock investment decision-making environment into consideration. The MASISS framework consists of mobile agent layer, business application layer, and resource layer shown in Figure 1. The mobile agent layer includes communication manager, agent gateway, and mobile server channel. The business application layer consists of service manager and a data mining engine. The resource layer includes a user subscriber database, a stock price historical database, and a category stock association rule database. The following sections explain various layers of MASISS.

3.1 Mobile agent layer

This agent layer, supported by Tahiti, of IBM's Tokyo laboratory, provides an application interface to model complicated agent behavior. It can create, clone, retract, dispose, and dispatch mobile agent. The agent layer has two major classes, AgletProxy and AgletContext. The AgletProxy is a proxy server that sends the client requests to the remote server and brings back the results after completing the assignment. The communication between agents can only pass through the designated functions. Hence the Agletproxy is like a protective umbrella that provides the transparency of objects. The AgletContext provides an implementation environment for the Tahiti platform. When a mobile investor's request is sent to a remote Tahiti server, it is decomposed into several message streams by its own AgletContext of the local Tahiti server and then sent to the AgletProxy of the remote Tahiti server. The message streams are then recomposed by the AgletContext of the remote Tahiti server. Figure 2 shows the detailed process of agent activities.

In addition, the mobile agent layer is further divided into three modules: communication manager, agent gateway, and mobile server channel. These modules are described as below.

- **Communication manager**: The communication manager is a software agent situated between wireless equipment and a wired network. The communication manager manages all system connections and communication with agent gateways. The system will automatically generate a user profile data record whenever a user uses a MIDlet download service channel via the communication manager, and will use Java Servlets and a Common Gateway Interface (CGI) to record the identity of the user profile and communicate the agent gateway's attributes and address.

- **Agent gateway**: The agent gateway provides an interface with the communication manager and establishing and updating user profiles. The agent gateway saves data to the user profiles database. In order to maintain security, the user is not permitted to directly access the user profiles database. The agent gateway can also implement the mobile agent platform, and this platform can be implemented effectively in a heterogeneous server environment – which is a key feature of mobile agents.

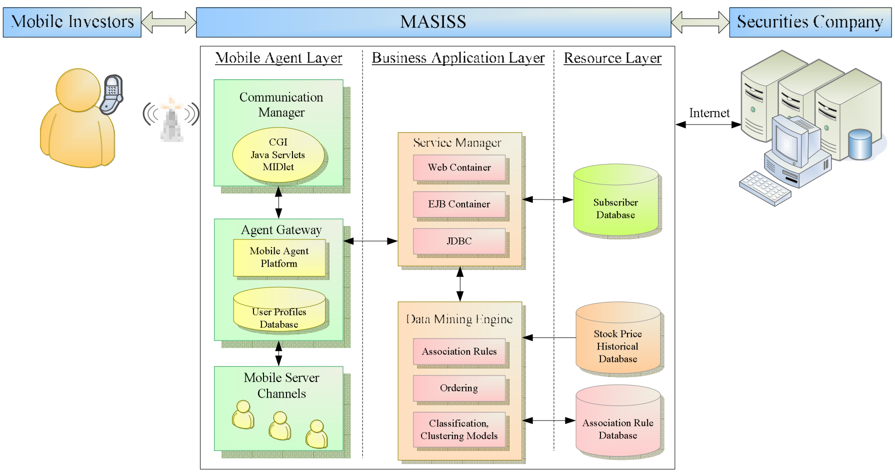

Figure 1. Mobile Agent-Based Stock Intermediary Service System (MASISS) Framework

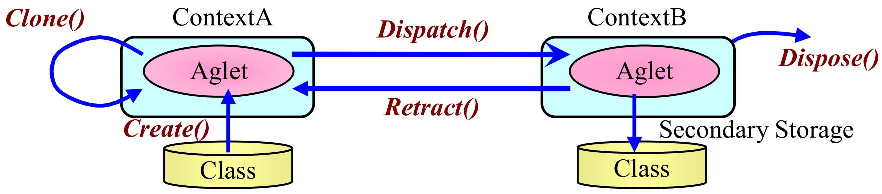

Figure 2. The Detailed Process of Agent Activities

✧ **Mobile server channels:** Mobile server channels satisfy users' multifaceted investment needs, such as analysis of arbitrary stocks, market analysis, intelligent stock selection, and category linkage analysis. A mobile server channel relies on the agent gateway to perform identification and accept user needs.

3.2 Business application layer

The business application layer is situated between the mobile agent layer and resource layer. It consists of service manager to provide services to investors, and data mining engine to find optimal association rule.

✧ **Service manager**: The service manager uses J2EE architecture to connect mobile agent layer, resource layer's subscriber database, and business application layer's data mining engine (Fig. 3). The J2EE architecture mainly consists of Web Container, EJB (Enterprise JavaBeans) Container, and JDBC (Java Database Connectivity). The Web Container includes JSP, Servlet, and JavaBean, to interact with the mobile agent and EJB Container. EJB container includes eSession Bean, Entity Bean, and Message Driven Bean. As a consequence, the service manager can push association rules obtained by the data mining engine to an investor's wireless handheld device at the client layer via the mobile agent in the mobile agent platform.

✧

✧ **Data mining engine:** The data mining engine mainly provides association rules function and application. The focus of the MASISS uses Apriori algorithm (Agrawal and Srikant, 1994) shown below to find optimal association rules.

✧

1. Determine the minimum support threshold.
2. Delete those candidate itemsets appearing less often than an itemset with minimum support threshold, and generation of frequent itemsets.
3. Join the mutual products in accordance with the generated set to create larger candidate itemsets. Loop Step 2 until no candidate itemset can be generated.
4. Find the itemset with the best confidence value among high frequent itemsets, and derive the optimal association rules.

Figure 4 shows an example. Assume the minimum support threshold is 2 and the trading database (TDB) contains four items of trade data, as well as five "candidate itemsets" {A}, {B}, {C}, {D}, and {E}. The degree of support (C1) of these five itemsets is obtained. The four "frequent itemsets" (L1) - {A}, {B}, {C} and {E} are obtained after deleting {D} with less than the minimum support threshold. Join the mutual products of L1 to generate "candidate itemsets" (C2) and yield their degree of support. Pruning all itemsets among C2 with less than the minimum degree of support will leave four "frequent itemsets" (L2) -

{A,C}, {B,C}, {B,E}, and {C,E}. Finally, all frequent itemsets (L3) are obtained by repeating the foregoing steps.

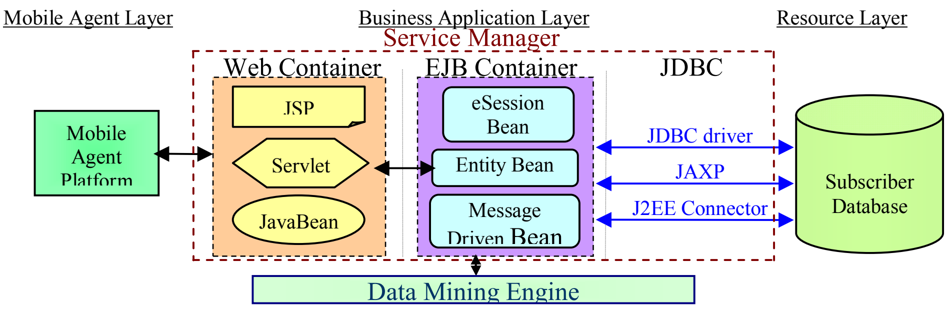

Figure 3. Multi-layer Business Application Architecture

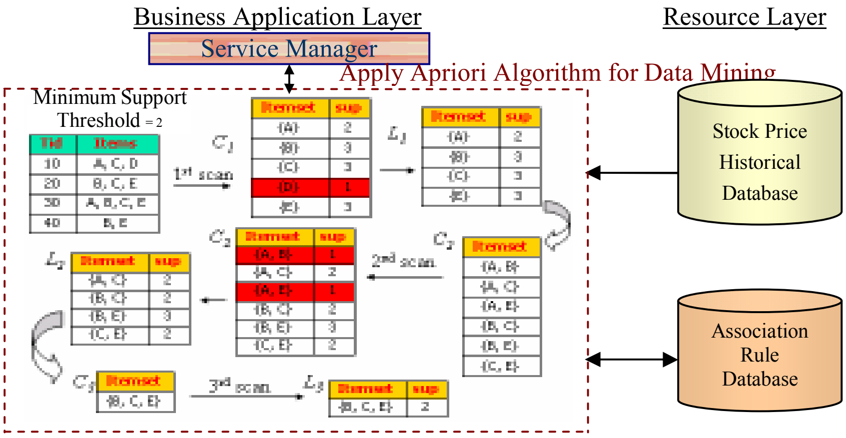

Figure 4. Application Data Mining Engine Employing Apriori Algorithm Procedures

3. 3 Resource layer

The resource layer mainly manages the resources stored in databases to support various applications and interactions in business application layer. It includes subscriber database, stock price historical database and application rule database.

- **Subscriber database**

The subscriber database accesses user subscription channel data and user profile records. The business application layer's service manager transmits association rules derived from the records in this database to the subscribing user.

- **Stock price historical database**

Stock price historical database accesses various raw stock market data. This raw data is the basic data source used for association rules analysis by the business application layer's data mining engine.

✧ **Association rule database**

The association rule database takes the subscriber database and stock price historical database as its data sources. This database compiles an association rule data warehouse from the enterprise's heterogeneous databases distributed at various locations. It then integrates, unifies, and summarizes information from various heterogeneous data sources, and generates data marts or data cubes. The database further uses Online Analytical Processing (OLAP) to perform effective online query and analysis. Finally, the business application layer's data mining engine performs analysis using the Apriori algorithm, and returns the resulting patterns or relationships to the association rule database or provides them to the business application layer's service manager for query purposes (Fig. 5).

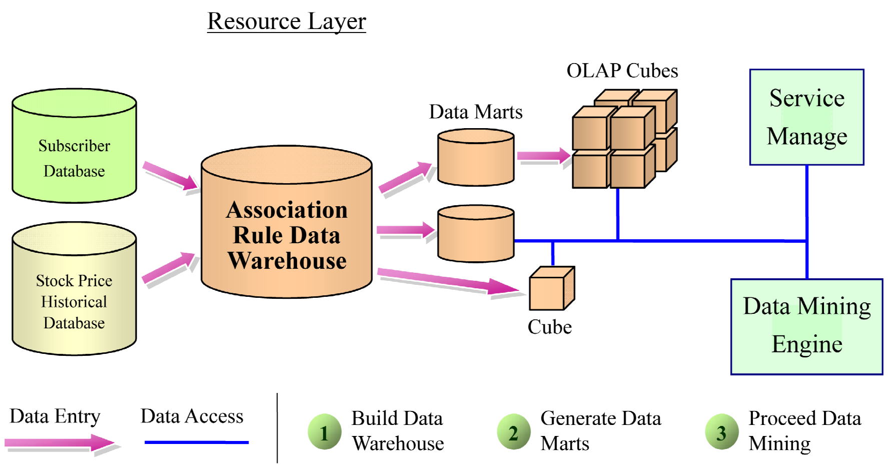

Figure 5. Association Rule Database Employing a Data Warehouse Architecture

4. System Evaluation of MASISS

The MASISS assessment process consisted of: (1) Obtaining and organizing the raw stock price data, (2) Using Apriori algorithm to find association rules, (3) Receiving information on investor's mobile phone, (4) Testing of feasibility of association rules. The following is a detailed description of these steps.

4.1 Obtaining and organizing the raw stock price data

Raw stock price data was obtained from "Intelligence Winner 2000" system of Infotimes Company in Taiwan. The selected stock category was "mobile phone parts and components concept stocks", which contained nine stocks 2393 - Everlight Electronics Co., Ltd. (A), 2402 - Ichia Technologies Inc. Technology (B), 2439 - Merry Electronics (C), 2448 - Epistar Corp (D), 2457 - Phihong (E), 3007 - Greenpoint (F), 3031 - Bright LED (G), 6168 - Harvatech (H), and 6285 - Wistron (I). The selected time period was from March 22, 2005 to April 22, 2005; data frequency was set as one day, and there was a total of 23 data sets and 22 stock price change data sets (Table 1). Then, find the stocks that rose for each day (Table 2). Any rising items in Table 2 (column I) that were blank or had only one transaction record were deleted because we only considered the condition of "concurrently rising" records.

Table 1. Summarized Stock Price Changes

ID	A	B	C	D	E	F	G	H	I
0323	1.6	0.35	0.9	3	-0.15	3	2.25	2.3	0
0324	0.6	0.3	-0.4	-0.25	0.3	8	-0.25	0.1	-2.3
0325	-0.6	1	2	-0.05	0.2	2.5	0.4	0	-3.3
0328	-0.15	0.9	5.6	0.9	0.15	0	1.9	0.1	-0.4
0329	-1.65	-2.1	-1.9	-0.8	-0.55	-6	-1	-0.8	-2.5
0330	-0.1	0.15	-1.2	-0.2	0.1	0.5	-0.65	0	-2.4
0331	1	0.05	-5.8	1.1	0.1	2	2.45	1.5	4.7
0401	0.55	-0.7	-2.8	0.2	-0.15	-0.5	0.6	-0.05	1.1
0404	-0.45	-0.5	0.1	-0.4	0.05	-1	-2.2	-1	0
0406	0.6	-0.35	1	0.7	-0.15	0	0.6	2.45	-1.2
0407	1.55	-2.05	-1.5	2.5	-0.05	-4	0.4	-0.45	-0.5
0408	0.55	-0.9	-2	0	0.1	2.5	0.9	-0.4	2.3
0411	0.1	0.65	-0.5	0.1	0	-3.5	1.2	0	-1.8
0412	-0.3	-0.1	1.2	-0.5	0	-0.5	0	-0.45	0.1
0413	1.05	-0.3	0.9	2.4	-0.15	2	0.45	0.6	0.4
0414	-0.05	0.05	-1.1	-0.1	-0.1	2	0.15	-0.3	-0.7
0415	0.25	0.05	-0.7	0.1	-0.35	0	0.3	-1	-0.3
0418	-1.65	-1.6	-1.8	-1.4	-0.65	-7	-2.05	-2.5	-5
0419	1.1	0.4	1.5	1	0.25	-0.5	1.3	0.1	1.5
0420	-1.75	-2.45	-0.5	-3.5	-0.22	-7	-2.55	-2.35	-4
0421	0.45	0.2	-0.3	0.4	-0.03	4	0.85	-0.75	3.3
0422	0.25	-0.2	0.7	0	-0.11	0.5	-0.65	-0.15	1.3

Table 2. Data Summary

ID	List of Item_IDs
0323	ABCDFGH
0324	ABEFH
0325	BCEFG
0328	BCDEGH
0329	
0330	BEF
0331	ABDEFGHI
0401	ADGI
0404	CE
0406	ACDGH
0407	ADG
0408	AEFGI
0411	ABDG
0412	CI
0413	ACDFGHI
0414	BFG
0415	ABDG
0418	
0419	ABCDEGHI
0420	
0421	ABDFGI
0422	ACFI

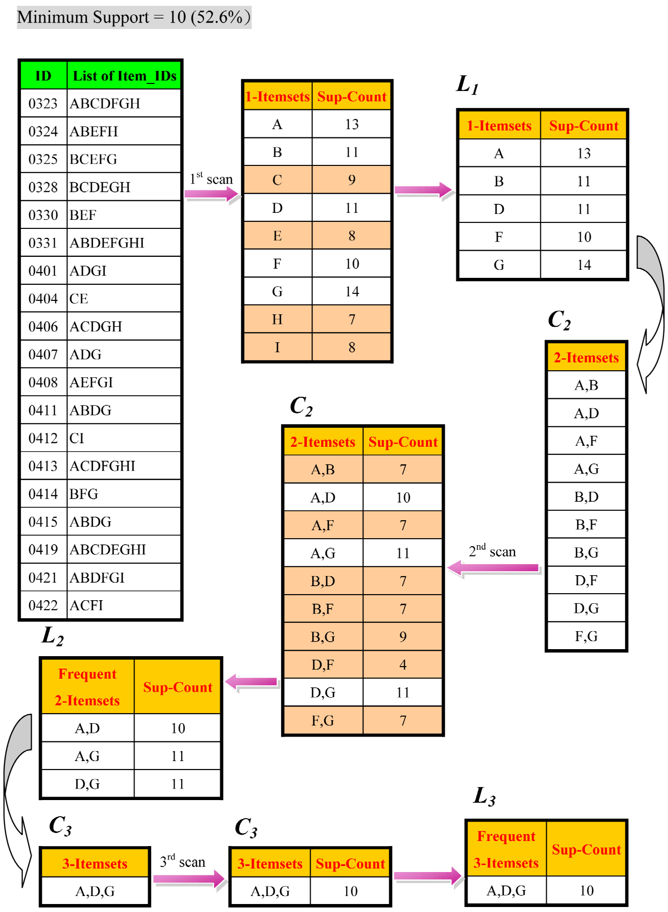

Figure 6. Association Rule Data Mining Using the Apriori Algorithm

4.2 Using Apriori algorithm to find association rules

Use of the Apriori algorithm to perform data mining yielded the final itemsets listed in Fig. 6. Table 3 specifies all association rules on the basis of the minimum support and confidence threshold. As an example, the meanings of the optimal association rules in Table 3 – "D => G" and "A D => G" - are as follows:

- ✧ Association rule "D => G": The probability of stock G simultaneously rising whenever stock D rises is 100% during the time period.
- ✧ Association rules "A D => G": The probability of stock G simultaneously rising whenever stock A and stock D both rise is 100% during the time period.

Table 3. Support and Confidence of Association Rules

Association Rules	Support	Confidence	Association Rules	Support	Confidence
$A \Rightarrow D$	52.6%	76.9%	$A \Rightarrow D \cup G$	52.6%	96.9%
$D \Rightarrow A$	52.6%	90.9%	$D \Rightarrow A \cup G$	52.6%	90.9%
$A \Rightarrow G$	57.9%	84.6%	$G \Rightarrow A \cup D$	52.6%	71.4%
$G \Rightarrow A$	57.9%	78.6%	$A \cup D \Rightarrow G$	52.6%	100%
$D \Rightarrow G$	57.9%	100%	$A \cup G \Rightarrow D$	52.6%	90.9%
$G \Rightarrow D$	57.9%	78.6%	$D \cup G \Rightarrow A$	52.6%	90.9%

4.3 Receiving information on investor's mobile phone

Investors can obtain stock analysis services by subscribing to service channels, and can regularly obtain reports on analysis of stock category association rules. If an investor wishes to obtain information on some other stock category, he/she can enter the "stock category analysis" function (Fig. 7 (a)), input the stock category code, and select a data period (Fig. 7 (b)). When the service manager receives the request, it will first determine whether there is data in the association rule database. If there is no relevant data, the service manager will immediately use data mining engine to perform association rules analysis, and send the results of analysis back to the investor's mobile phone via a mobile agent (Fig. 7 (c)).

(a) Select Services (b) Input Stock Category/Data Period (c) Receive the Results of Analysis

Figure 7. Operating Interface of Investor's Mobile Phone

4.4 Testing of feasibility of association rules

This study used the association rule "D => G" to test feasibility. The time period was from April 25 to May 6. The price changes of these nine stocks are shown in Table 4. Four valid data items that remain after data items for stock D's failure to rise were deleted. Three of these four data items comply with the results of our analysis. This result indicates that the probability of 3031 rising when 2448 rises is 75%. We can also see that stock G uniformly fell whenever stock D fell, which proves that there is a high degree of association between the prices of stock D and stock G.

Table 4. Subsequent Stock Price Change Data

ID	A	B	C	D	E	F	G	H	I
0425	0.85	-1.3	0.1	-1.3	-0.22	-3	-1.8	-0.55	-0.2
0426	0.3	0.15	0.6	1.35	0.1	-1.5	0.6	0.3	0
0427	-0.8	-0.75	-0.7	-0.85	-0.12	-0.5	-1.6	-0.4	-1.4
0428	0	-0.1	0.7	0	0.07	3	0.7	0.1	-2
0429	-2.2	-1.25	-1.3	-2.5	-0.15	-2.5	-0.1	0.05	-0.6
0503	0.6	-0.95	0	0.6	0.13	3	-0.2	0.45	4
0504	-0.6	-2.1	0	-0.1	-0.13	-2.5	-0.4	-0.5	-1.1
0505	2.75	-1.1	1.3	3.15	0.2	3	2.35	1.7	1.1
0506	1.85	0.65	1.9	2.95	0.03	1	1.55	0.25	2.5

5. Conclusions

Mobile stock investment decision is already an inevitable trend in recent year. It provides important judgments to help investors to make investment decisions in the mobile commerce environments. However, seldom researches are focused on the intermediary services system for both stock investors and securities companies based on mobile agent perspective. This study proposes a Mobile Agent-Base Stock Intermediary Services System (MASISS) framework, which allows mobile agents to operate independently and asynchronously, and significantly reduces the bandwidth usage over the network. Hence, it provides the advantages of autonomy and independence to the stock investors and stock companies. This is particularly useful if the agents are implemented under a heterogeneous environment; the framework allows the minimum system integration effort to implement the agent-based system. Furthermore, this study shows the ease of aggregating investors' stock order for decreasing transaction cost by using intermediary mobile agents. The framework allows mobile investors to possess the mobility and save a great deal of time and money, if comparing our framework with using mobile push personalized information matching based on the mobile users requirements via publish and subscribe channel to wireless handheld devices. The mining engine also effectively extract investment rule to provide decision makers the most reliable information. Finally, the Java-based mobile agent system can be implemented on heterogeneous platforms and interoperate with cross-platform applications via wireless handheld devices in practice.

References

[1] G. P. Picco, Mobile agents: an introduction, Microprocessors and Microsystems 25 (2001) 65-74.
[2] S. S. Manvi, P. Venkataram, Applications of agent technology in comm.: a review, Comp. Comm. 27 (2004) 1493–1508.
[3] D. B. Lange, M. Oshima, Seven good reasons for mobile agents, Comm. of the ACM 42(3) (1999) 88–99.

[4] J. Cao, X. Wang, S. K. Das, A framework of using cooperating mobile agents to achieve load sharing in distributed web server groups, Future Generation Comp. Sys. 20 (2004) 591–603.
[5] R. Agrawal, R. Srikant, Fast algorithms for mining association rules, Proc. of 20th International Conf. on Very Large Data Bases, Santiago, Chile, (1994) 487-499.
[6] T. C. Du, E. Y. Li, E. Wei, Mobile agents for a brokering service in the electronic marketplace, Decision Support Systems 39 (2005) 371-383.
[7] S. H. Chen, C.C. Liao Agent-based computational modeling of the stock price-volume relation, Information Sciences 170 (2005) 75-100.
[8] M. J. Wooldrige, N. R. Jennings, Agent theories, architectures and languages: a survey, Lecture Notes in Computer Science 890 (1995) 1-39.
[9] D. H. Shih, S. Y. Huang, D. C. Yen , A new reverse auction agent system for m-commerce using mobile agents, Computer Standards & Interfaces 27 (2005) 383-395.
[10] T. Sandholm, Q. Huai, Nomad: mobile-agent system for an internet-based auction house, IEEE internet Computing (2000) 80-86.
[11] P. Dasgupta, N. Narasimhan, L. E. Moser. P. M. Melliar-Smith, MAgNET: mobile agents for networked electronic trading, IEEE Trans. on Knowledge and Data Engineering 11 (4) (1999) 509-525.
[12] J.J. Jung, G. S. Jo, Brokerage between buyer and seller agents using constraint satisfaction problem models, Decision Support Systems 28 (4) (2000) 293-304.
[13] K. G. Zerfridis, H. D. Karaza, Brute force web search for wireless devices using mobile agents, J. of Sys. and Software 69 (2004) 195-206.
[14] M. J. O'Grady, G. M. P. O'Hare, C. Sas, Mobile agents for mobile tourists: a user evaluation of Gulliver's Genie, Interacting with Comp. 17 (2005) 343-366.
[15] T. C. Du, E. Y. Li, A. P. Chang, Mobile agents in distributed network management, Comm. of the ACM 46 (7) (2003) 127-132.
[16] D. Chess, C. Harrison, A. Mobile agents: are they a good idea? IBM Research Division, research report, (1995).
[17] J. T. S. Quah, G. L. Lim, Push selling-multicast messages to wireless devices based on the publish/subscribe model, Electronic Commerce Research and Applications 1 (2002) 235-246.
[18] T. Hiroyuki, K. Koji, I. Yoshihide, Y. Kenji, A Takayoshi, WorkWeb system-multi-workflow management with a multi-agent system, Proc. of ACM SIGGROUP Conf. on Supporting Group Work: the Integration Challenge, Phoenix, Arizona, USA, (1997) 299 – 308.