Available online at www.sciencedirect.com
SCIENCE DIRECT®
Knowledge-Based Systems 18 (2005) 257–266
Knowledge-Based SYSTEMS
www.elsevier.com/locate/knosys

The design of intelligent workflow monitoring with agent technology☆

Minhong Wang*, Huaiqing Wang, Dongming Xu[1]

Department of Information Systems, City University of Hong Kong, Hong Kong, China

Received 20 October 2002; accepted 6 April 2004
Available online 4 May 2005

Abstract

In recent years, workflow technology has been widely used in business process management. With the increased complexity, uncertainty and risks in business operations, workflow monitoring is gaining growing attention in business process controlling and supervision. However, monitoring functions provided in traditional workflow systems lack flexibility, and provide little support for managing complex and dynamical changes in business process. In this paper, we propose a novel workflow monitoring approach, in which various intelligent agents work together to perform flexible monitoring tasks in an autonomous and collaborative way. By using customized monitoring plan and proactive monitoring process, the workflow monitoring activities can be executed flexibly and efficiently. The application of intelligent agents for such flexible, adaptive and collaborative workflow monitoring is investigated through an intelligent monitoring system in securities trading.

Keywords: Workflow monitoring; Workflow management system; Intelligent agent

1. Introduction

Workflow management is a promising technology aiming at the automation of business processes to improve the speed and efficiency of an organization. In recent years, workflow management systems (WfMS) have been widely used in business process controlling and monitoring. With the increased complexity, uncertainty and risk in business operations, there is an increased demand on flexible and dynamic workflow management [4]. Workflow monitoring, an important function in workflow management systems, has gained growing attention in business process management. The historical data of workflow instance execution provide a valuable source for active monitoring of the current state or performance of workflow instances.

However, monitoring functions provided in traditional workflow systems lack choices and flexibility to deliver relevant information to appropriate persons. In Workflow Reference Model provided by the Workflow Management Coalition, standard monitoring APIs are defined to enable the analysis of automated business processes through the analysis of the logged audit trail data [25]. However, such API supports monitoring solutions limited to a few built-in options and process-relevant events, and there are some problems such as information overloaded or information not delivered exactly to the user who need it [1].

In this paper, we propose a flexible workflow monitoring approach, in which a society of intelligent agents work autonomously and collaboratively to perform monitoring tasks. Agent technology with its properties of autonomy, reactivity, and pro-activity provides an extension and alternative to business process management. By the aid of intelligent agents, our system can execute customized monitoring tasks for different users based on their individual monitoring requests. Furthermore, with the proactive property of intelligent agents, the monitoring process can be adjusted based on the information generated during monitoring activities. By using customized monitoring plan and proactive monitoring process, the workflow monitoring activities can be executed flexibly and efficiently. The application of intelligent agents for such flexible and adaptive workflow monitoring is investigated through an intelligent monitoring system in securities trading.

☆ The paper is supported by Strategic Research Grant (7001141 and 7001309) from City University of Hong Kong.
* Corresponding author. Tel.: +852 2788 9962; fax: +852 2788 8535.
E-mail address: magwang@cityu.edu.hk (M. Wang).
[1] Present address: UQ Business School, University of Queensland, Australia.

0950-7051/$ - see front matter
doi:10.1016/j.knosys.2004.04.012

The remainder of the paper is organized as follows. Section 2 briefly reviews the relevant literatures in workflow management, intelligent agents, and other related fields. Section 3 illustrates our intelligent monitoring approach that consists in customized monitoring plan and proactive monitoring process. Based on this approach, an example of intelligent monitoring system for securities trading is elaborated in Section 4. In this section, possible errors occurring in securities trading are discussed; a multi-agent framework for monitoring of securities trading is proposed, as well as the agent hierarchy with their communication and collaboration in monitoring activities. Finally, Section 5 concludes with some advantages and limitations of our agents supported workflow monitoring approach.

2. Background

2.1. Workflow management and workflow monitoring

A workflow management system (WfMS) is the software that automates the co-ordination and control of tasks during business process execution. The workflow approach helps to separate the business logic represented by business process from the underlying information systems that support the process. This separation allows business processes to be designed without requiring major changes to be made to the underlying computing infrastructure [12]. Nowadays, workflow management has become a promising technology aiming at the automation of business processes to improve the speed and efficiency of an organization. The success of workflow paradigm is based on its ability to support modeling, simulation, automated execution, and monitoring of processes in an environment that is distributed, heterogeneous, and only partially automated. While workflow technology has seen an explosion of interest and advances in recent years, numerous technical challenges have been addressed to provide flexible workflow management systems required by complex and dynamic application domains [4].

Workflow management systems enable the exact and timely analysis of automated business processes through the analysis of the logged audit trail data [1]. Active monitoring of the current state of workflow instances can serve numerous purposes, such as the generation of exception reports for overdue work items or early warning reports for potential workflow problems. Passive monitoring upon request can deliver status information about running workflow instances, e.g. for answering a customer inquiry about the status of an order. In Workflow Reference Model, a standardized set of interfaces and data interchange formats is provided to achieve interoperability between workflow products. As outlined in Fig. 1, standard monitoring APIs are supported via Interface 5, which specifies the elementary information a workflow management system should record about the execution of workflow instances [25]. Though the detail of this interface is for further study, it is feasible for the management application to take on some management functions, such as user management, role management, resource control, audit management, process supervision, and so on. The process supervision may contain opening/closing a process or activity instances query, setting optional filter criteria, fetching details of process instances or activity instances, changing the state of process or activity instances, termination of process instances, etc. However, such API supports monitoring solutions limited to a few built-in options and process-relevant events, and there are some problems such as information overloaded or information not delivered exactly to the user who needs it [1].

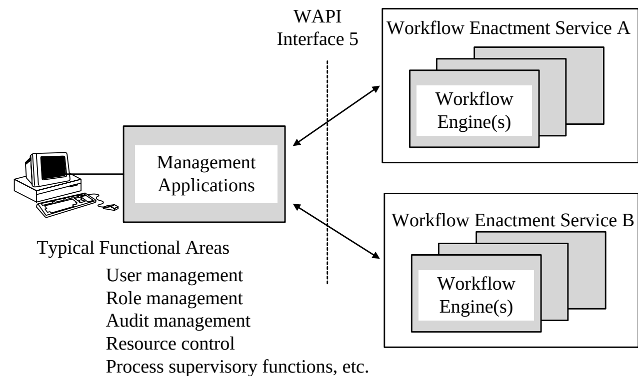

Fig. 1. Interface 5 in the workflow reference model.

2.2. Intelligent agents in workflow management

In recent years, the concept of agent has become increasingly important in both artificial intelligence and computer science. The term of intelligent agent is used to denote a software-based computer system that enjoys the following properties: (a) autonomy (agents operate without the direct intervention of humans); (b) social ability (agents communicate with other agents); (c) reactivity (agents perceive their environment and respond in a timely fashion to changes that occur in it); (d) pro-activity (agents do not simply act in response to their environment, they are able to exhibit goal-directed behavior by taking the initiative) [23,24]. A generic agent has a set of goals (intentions), certain capabilities to perform actions, and some knowledge (or beliefs) about its environment. To achieve its goals, an agent needs to use its knowledge to reason about its environment (as well as behaviors of other agents), to generate plans and to execute these plans. A multi-agent system consists of a group of agents, interacting with one another to collectively achieve their goals. By drawing on other agents' knowledge and capabilities, agents can overcome their inherent bounds of intelligence.

Traditional workflow systems have certain limitations such as replying on central control, lack of reactivity, semantics, resource management and interoperation. Intelligent agent technology is among the ones that can benefit the workflow technology [9,12,26]. Agent-based workflow approach can provide highly reusable component-based workflow architecture as well as negotiation ability and the capability to adapt to dynamic changes in the environment. The benefits of applying agent technology to workflow management include distributed system architecture, easy interaction, resource management, reactivity to changes, interoperation among heterogeneous systems, and intelligent decision making. In recent years, the integration of workflow and agent technology has attracted considerable interests in research to support business processes management in distributed, dynamic and unpredictable environment. O'Brien and Wiegand presented an agent-based Process Management System (APMS) architecture that extends workflow with the abilities to anticipate process requirements, to resource processes dynamically, and to adapt to exceptions [12]. Chen et al. developed a Java-based dynamic agent infrastructure for E-Commerce automation, which supports dynamic behavior modification of agents, a significant difference from other agent platforms [3]. Zhuge et al. proposed a simulation-based development framework for establishing virtual organizations by using multi-level agents to perform relevant tasks according to an overall workflow [27]. In these applications, agent technology provides an extension and alternative to business process management problems.

2.3. Other related research fields

This research has also related to other fields such as plan recognition, intention inference, adaptive information and automation management, etc. Plan recognition is the process of inferring the goal of an agent from observations of an agent's actions [6]. The earliest work in plan recognition was rule-based; researchers attempted to come up with inference rules that would capture the nature of plan recognition. Goldman et al. developed a new theory of plan recognition based on abductive and probabilistic logic, which permits solutions to a wider class of plan recognition problems [6]. Charniak and Goldman argued that the problem of plan recognition is largely a problem of inference under conditions of uncertainty, and presented an approach to the plan recognition based on Bayesian probability theory [2].

It has been noticed that monitoring difficulties may be characteristics of conventional, 'static' automation, in which function allocation between the human operator and automated systems remains fixed. In contrast, adaptive systems are thought to be less susceptible to automation-induced difficulties in monitoring [13]. A distinctive feature of an adaptive system is an explicit user model that represents user knowledge, goals, interests, and other features that enable the system to distinguish among different users. An adaptive system collects data for the user model from various sources that can include implicitly observing user interaction and explicitly requesting direct input from the user. A number of topics have been addressed in this area, such as model-based interface design, intelligent interfaces that adapt to users' changing resource limitation, how to implement interfaces that support collaboration [14,16]. Guerlain and Bullemer argued that monitoring activities in a process control environment are quite unique depending on the current situation and operator's current understanding of that situation. There is a limited set of conditions that can be pre-defined in a tool that allow operators to set up their own monitoring tasks. They described an interface design concept that would allow operators in a process control plant to initiate monitoring events based on their current diagnostic needs, so-called 'user-initiated notification' [7].

3. Intelligent monitoring solution

As mentioned before, workflow monitoring supported in traditional workflow systems lacks choices and flexibility to deliver relevant information to appropriate persons. It will affect the performance of process control in manufacture and business activities, especially in today's dynamic environment. In this paper, we are to apply the technology of intelligent agents into workflow monitoring, with a view to provide more flexibility and intelligence in current workflow monitoring approaches. Based on our past research and experimental results, some properties of intelligent agents, such as reactive and proactive behaviors, are directly applicable to tackling abnormal transactions in a systematic and goal-oriented manner [17,18,21,22]. By the aid of intelligent agents, user-initiated planning for monitoring activities is proposed to make the right information delivered to the right person. Furthermore, with the assistance of intelligent agents, proactive monitoring process is facilitated to make the monitoring process more flexible and efficient [20]. In other words, we will provide the system with some capabilities to control or adjust monitoring processes based on information from users or generated in the monitoring activities. The details of our approach are described as follows.

3.1. Customized monitoring plan

The support of flexibility in monitoring solution is a main concern of this research, which is achieved through the customization of monitoring plans. Different people may have different monitoring requirements on business monitoring. In our system, the User Agent is proposed to help people input their monitoring requests, while the Planning Agent can generate individual monitoring plans based on users' input. The skeletons for major monitoring activities in an organization are pre-defined in the knowledge base of the Planning Agent. A customized monitoring plan is generated through the integration of a monitoring skeleton and individual monitoring requests from the user. For instance, a user who is responsible for material supply in a manufacturer needs to monitor the fulfillment of purchase orders to ensure accurate supply of materials for production. By the aid of the User Agent, the user can choose an appropriate monitoring skeleton to monitor the status of purchase orders. After choosing this skeleton, the user can input his/her individual monitoring requests for this skeleton, such as monitoring interval, monitoring scope (e.g. orders for some important materials), critical monitoring items (e.g. orders suspended for 20 min, shipments delayed for several hours), and so on. Thus, the user has created an individual monitoring plan and will receive alert messages once a purchase order is in a critical status.

3.2. Proactive monitoring process

A monitoring process in our approach involves two phases: primary monitoring and further monitoring. Every monitoring event starts from the primary monitoring, which focuses on tracking primary items. It may automatically go on with further monitoring if some items in the primary phase are found critical or abnormal. The transition from the primary monitoring phase to the further monitoring phase is controlled by the Diagnostic Agent. Following the above example about purchase order monitoring, the monitoring process includes the primary monitoring on current status of purchase orders and further monitoring on detailed information of critical orders. If an order is in a critical status, a warning message with reports will be sent to the manager and a further monitoring process will be started to collect more information about this order, such as the current inventory level of this material, other possible suppliers for this material, etc. Such kind of proactive monitoring process may reduce the cost of monitoring activities with high efficiency. In our system, the Diagnostic Agent with its proactive property to facilitate is quite applicable to such flexible monitoring processes. The detailed design of this agent is described in Section 4.2.4.

4. An intelligent monitoring system of securities trading

In order to demonstrate our intelligent agents supported workflow monitoring approach described above, we have elaborated a multi-agent system, in which a society of intelligent agents work autonomously and collaboratively to perform monitoring tasks in financial securities trading. By using customized monitoring plan and proactive monitoring process, our system can support more flexibility and intelligence in monitoring activities for securities trading process. The details of this system are described below.

4.1. Domain analysis on securities trading

The trading of securities is a regulated process. The general lifecycle of securities transactions consists of several steps, including order placement, trade execution, trade agreement, trade clearance, settlement instruction, and payment and delivery [15]. In securities trading process, there are usually some errors or problems that may cause transaction risks and failures. Such problems not only increase processing costs but also damage service reputations [8]. One kind of error is related to trade records. Such errors usually occur by the trader when entering trade details to the trade system or when writing the details of a trade, or by middle or back office personnel inputting trades manually into the settlement system. For instance, the trader deals a trade at a price that is significantly different from the market price. Or, a trade has been captured with some unusual components, such as value date on a public holiday, incorrect calculation of trade cash value, missing some details, and so on. The consequence of failing to identify, investigate and resolve such errors early in the trade lifecycle may mean that resolution may well take considerably longer and risks increase. Another kind of error is related to trade agreement. Trade agreement is to gain agreement of the trade details between the parties to a trade soon after the trade is executed. In modern settlement systems, the issuance of trade confirmation can usually be automated. The issuer of trade confirmation hopes that the recipient will check the detail upon receipt; however, it is not always the case. Sometimes the counterparty cannot recognize the confirmation message, or did not act upon matching advice in a reasonable time frame, or failed to receive or reply the message due to transmission errors [8].

In order to manage such problems, the securities industry has made some efforts to automate the exception management in securities transactions [8,19]. With the increasing complexity, uncertainty and risks in securities transactions, there is an increased demand on flexible and dynamic transaction monitoring. Different managers may have different monitoring requirements. Some may focus on managing the errors in trade recording, some may concentrate on tracking trade agreement activities, and some would pay more attentions to payment activities. Furthermore, even for the same monitoring activity, the monitoring policy may need adjustment according to changes from business environment, business strategy, manager's perception, etc. For example, when monitoring trade price, the manager needs to decide how 'tight' to make the tolerance relative to current market price. Too tight may well result in many trades being held for validation, requiring much efforts to investigate and resolve the problems; too lose may result in incorrect prices being processed undetected. Another example, when specifying the monitoring scope, the manager can only choose the trades of large value, and he needs to decide how 'large' the value is specified for the monitoring scope. Too large may cause too many trades

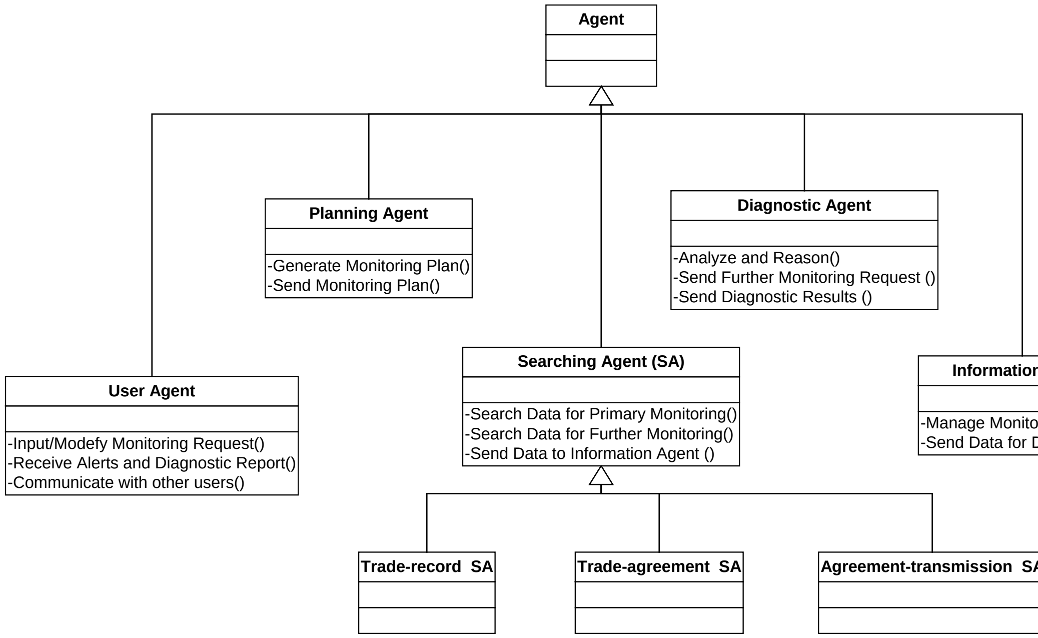

Fig. 2. Class diagram of intelligent agents in the monitoring system.

passed over without validation, and too small may bring too many monitoring tasks.

4.2. Intelligent agents for securities trading monitoring

Based on the errors and monitoring requirements in securities trading process described above, we have designed a multi-agent system for detection and identification of transaction errors with intelligence and flexibility. In a multi-agent system, cooperation among agents is an important issue, and the task distribution in the system has been considered as one of the main goals for this cooperation. In our system, a monitoring activity is decomposed into several sub-tasks, such as constructing a customized monitoring plan, searching required data, diagnosing the nature of problems, and generating diagnostic reports. After decomposition, such tasks can be assigned to various agents, i.e. Planning Agent, Searching Agent, and Diagnostic Agent. They are a collection of computational entities having their own problem-solving capabilities and able to interact in order to reach the overall goal of intelligent monitoring. Based on such analysis, the intelligent agents in our system can be classified into several classes. The taxonomy of intelligent agents with specific behaviors is shown in Fig. 2. In a multi-agent system, it is essential to design a set of autonomous types of behavior, including reactive, proactive, and cooperative behavior, for the agent class to have actions based on its own goal and other agents' request [17]. The reactive behavior of an agent allows the agent to have actions based on other agents' requests; the proactive behavior enables an agent to act based on its goal; the cooperative behavior enables an agent to cooperate with other agents. The details of each agent are illustrated below.

4.2.1. User Agent

User agents act as effective bridges between users and computers. Such agents can make the human–computer interface more intuitive and encourage types of interactions that might be difficult to evoke with a conventional interface [18]. In our workflow monitoring system, the User Agent is proposed to help users to initiate their monitoring plans, receive warning messages, read diagnostic reports and communicate with other users. Based on the two kinds of major errors that occur in securities trading process, two kinds of manager can be deployed in the securities trading monitoring system, the trade-record manager and the trade-agreement manager. The former is responsible for keeping track on trade recording errors, while the latter is charged to track the trade agreement activities.

As an example outlined in Fig. 3, the trade-record manager is to initiate a monitoring plan through the User Agent. Firstly, the manager logged into the securities monitoring system and clicked 'Create monitoring plan' to start the planning activity. Secondly, in the pop-up information of monitoring skeleton, the manager may choose 'Monitoring trade record' as his/her monitoring skeleton and submitted it to the User Agent. Thirdly, based on the user's monitoring skeleton, the User Agent presented additional information of monitoring requests. The manager can specify the start time of monitoring events, monitoring interval, monitoring scope of trades (e.g. the trades of value more than 10,000 HK$, and the trades executed by telephone or manually). Moreover, the manager may specify critical items for monitoring activities. In this

Fig. 3. User Agent for trade-record manager.

example, when a trade is detected of significantly different price from market price, or with invalid account or invalid security, an error will be reported to the manger.

4.2.2. Planning agent

The Planning Agent is charged to process the user's monitoring requests received from the User Agent, and then generates a customized monitoring plan for the user. Some pre-defined monitoring skeletons are stored in the knowledge base of the Planning Agent, and can be presented to users through the User Agent. After the user selects his/her monitoring skeleton and inputs individual requests or options, all these inputs are sent to the Planning Agent. Thus, a customized monitoring plan is generated and subsequently distributed to Searching Agents, Information Agent, and Diagnostic Agent to perform their tasks for information collection and detection.

Following the example in Fig. 3, the Planning Agent may create an individual monitoring plan as outlined in Table 1. The monitoring plan consists of monitoring skeleton 'monitoring trade records', and related monitoring requests including start time "10/10/2003 09:00", monitoring interval "20 minutes", monitoring scope, and critical item. The monitoring scope is specified as trades with value more than 10,000 HKD or trades executed by telephone or manually. The critical item defines trades with 5% price difference, invalid account and invalid security name to be detected as errors.

Table 1
A monitoring plan of trade-record manager

```
Monitoring_plan_x1 (Monitoring_skeleton ("monitoring trade record"),
  Monitoring_request (Start_time ("10/10/2003 09:00"),
    Montoring_interval ("20 minutes"),
    Monitoring Scope (Trade_value (">10000HKD"),
        Trade_mode ("telephone", "manual")),
    Critical_item (Price ("5% difference"),
      Account ("invalid"),
      Security ("invalid"))))
```

4.2.3. Searching agent

The main task of Searching Agents is to collect relevant information from the workflow system or other related systems on behalf of users or other agents. In order to search non-local information, such Searching Agents should have the mobile capability to search related information through networks. They can access multiple, heterogeneous and geographically distributed information resources in corporate Intranets or Internet. In our securities monitoring system, there are three kinds of Searching Agent (SA), Trade-record SA, Trade-agreement SA and Agreement-transmission SA; each assists to collect relevant data from the real trading and settlement system or other related applications. The Trade-record SA collects the details of each trade to check recording errors; the Trade-agreement SA collects the information of trade matching status (e.g. matched, mismatched, unknown) to find trades that have not been agreed by trade parties in a specified time frame; Agreement-transmission SA collects transmission records of each trade agreement for checking their transmission status.

4.2.4. Diagnostic Agent

The role of the Diagnostic Agent is to make data analysis and reasoning based on some rules defined in monitoring plans. Such business rules form an important part of knowledge base of the agent to perform delegated tasks on the user's behalf [11]. Once the rules are understood, captured and represented in form of logic, they will serve as a basis for building intelligent agents to perform rational activities. Business rules specify a series of clear statement about the logic underlying business practice. Table 2 is an example of a business rule written in Jess (Java Expert system Shell). This rule is used to check if there is any important item missing in a trade record.

While designing rules for intelligent agents to perform monitoring activities, we may not merely focus on those rules to control monitoring and diagnosing activities, but also consider some business strategies to improve the efficiency of monitoring activities. As discussed in Section 3, we use the Diagnostic Agent with its proactive property to improve the efficiency of monitoring activities. Pro-activity is an important property of agents, which means agents do not only simply act in response to the environment but also can exhibit goal-related behaviors by taking

Table 2
A monitoring rule

```
(defrule check_missing_item
  (trade (Trade_id ?trade-id) (Security_id ?securitiy-id) (Price ?price)
  (Quantity ?quantity)
    (Custodian_account ?account) (Trade_time ?trade-time))
  ⇒
  (if (or (eq ?security-id nil) (eq ?price nil) (eq ?quantity nil) (eq ?account
  nil) (eq ?trade-time nil))
  then (assert (item_missing ?trade-id))))
```

the initiatives [24]. In our system, every monitoring event starts from the primary monitoring phase. Based on the result from this phase, the Diagnostic Agent will decide whether to go on with a further monitoring process. This kind of decision is also made by the rule in the Diagnostic Agent. For example, if the Diagnostic Agent finds a trade that has not been agreed by trade parties in a reasonable time frame, the further monitoring activity will be started to check if there is any transmission error for this trade agreement. This kind of diagnostic knowledge is stored in the knowledge base of the Diagnostic Agent to support the goal-related behavior of this agent.

4.2.5. Information Agent

Information agents are used to assist users in information retrieval, information filtering, or other information manipulation. In our system, the Information Agent is responsible for the information manipulation for monitoring activities. The information agent will filter, store, and reorganize information from Searching Agents, and then send them to the Diagnostic Agent for further analysis and reasoning. This agent plays an important role in our approach. Although there is no need for centralized storage of all knowledge regarding monitoring activities, there could be one consistent knowledge repository that maintains and integrates all information related to the monitoring and analysis of tasks. In this way, the various agents that make up the system can exchange knowledge regarding entities involved and deal with exceptions in a collaborative manner.

4.2.6. Agent structure

The behavior of an agent is based on an internal model of the agent consisting of a knowledge base, operational facilities, and a correspondence between the external application domains. Generally, developing of an agent considers an agent knowledge base, its operational facilities and its external interface. Knowledge is required by each agent to perform its internal and external activities. It consists of knowledge for particular tasks, resource status information, information about other agents, etc. The operational facilities execute different functions and provide collaboration with other agents; they are the central control and action part of an agent. A rule engine is usually an important operational facility, which provides a means for applying simple, rule-based reasoning to emergence of new facts in the agent's world and for using this reasoning capability to decide what the agent should do next. The external interface envelops an agent and provides access to it via a well-defined interface, and it is also the primary conduit for communication between agents. The example of agent structure is presented in Fig. 4, from which we can take a look at the internal model of the Diagnostic Agent.

4.3. System architecture

Our securities monitoring system is to track and identify errors in securities transactions. We can reengineer current trading applications to support intelligent monitoring functions, or develop an independent monitoring system to link with existing applications. In this research, we try to fundamentally use internal resources to build software capabilities to interact with legacy systems. Relevant data are extracted from existing trading systems into the monitoring system to perform monitoring activities on securities transactions; some alerts and reports for identified errors will be sent back to trading systems for repair actions.

Based on the deployment of intelligent agents in Section 4.2, we outline the framework of our

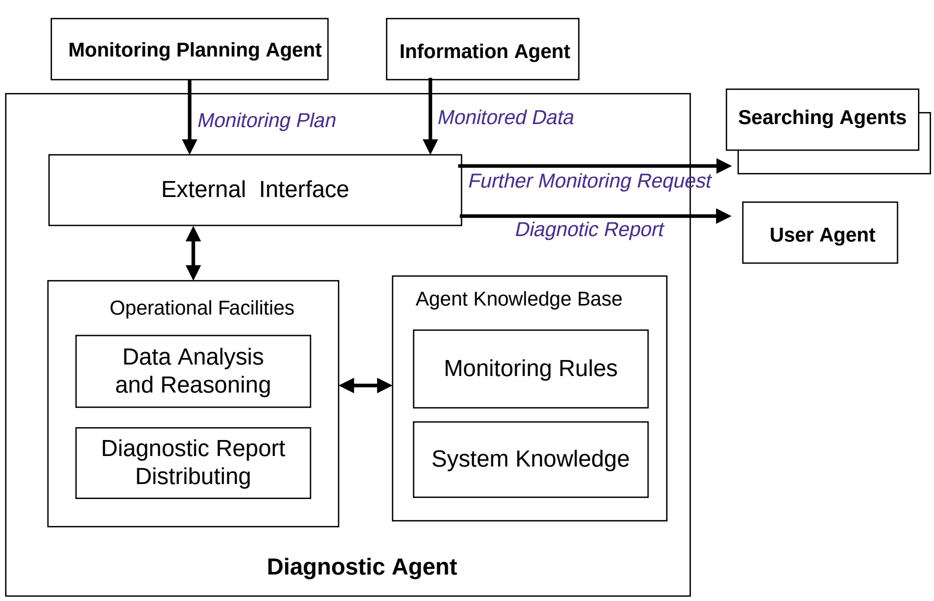

Fig. 4. The structure of the Diagnostic Agent.

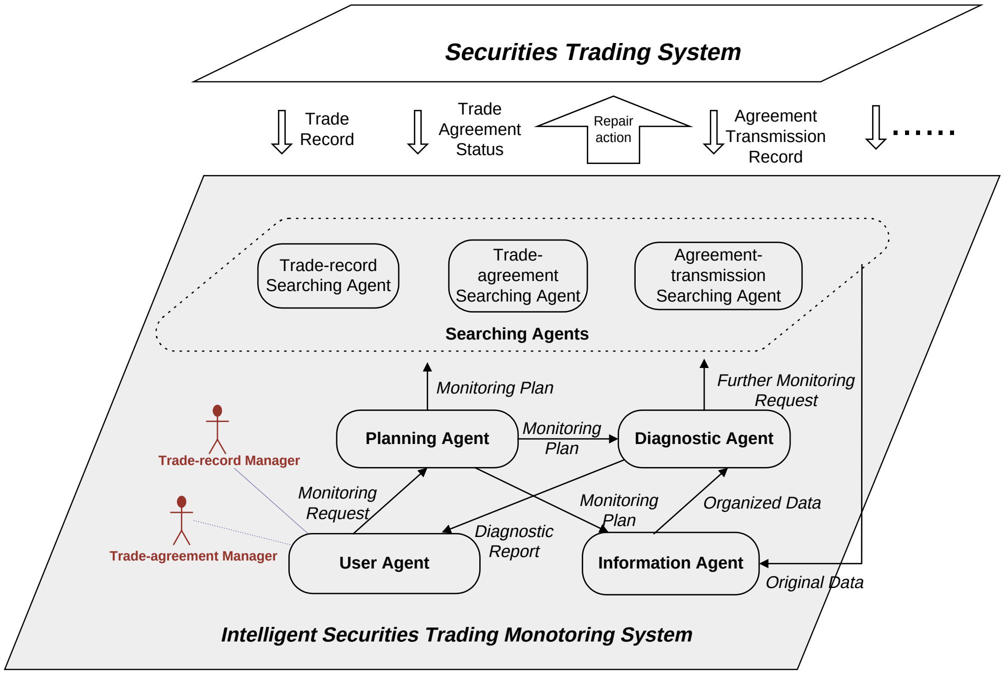

Fig. 5. Architecture of intelligent securities trading monitoring system.

multi-agent-based monitoring system in Fig. 5, in which a society of intelligent agents is applied to provide a set of flexible monitoring functionalities for securities trading. This framework provides the conceptual basis for thinking about the multi-agent-based flexible and adaptive monitoring system.

4.3.1. Agent communication

Interaction among agents, an important aspect on research of multi-agent system, is set up on lower-level data communication as well as control information with semantic and knowledge. The most popular language for agent communication is Knowledge Query and Manipulation Language (KQML). Recently, there are some researches focusing on the use of XML (Extensible Markup Language) in agent communication [5]. XML is a meta-language, that is, a language used to describe a language. It enables the definition of customized markup languages for different classes of documents. In our system, agents will send and receive information through XML encoded message with KQML-like format. The use of XML and KQML enables software agents to understand the contents of messages correctly and consistently.

4.3.2. Agent collaboration

Communication and collaboration are the different layers of view of interaction. Communication allows participants in the decision process to share information (this involves networking infrastructures), and collaboration allows participants to collaboratively update some shared set of decisions (this involves support for tele-conferencing, etc.) [10]. Collaboration means more than just instantaneous communication, or total asset visibility or leveraging resources and the talents of experts from different fields. Collaboration requires two or more participants who contribute to a common task. A crucial point for successful collaboration is the manner in which individual work is related to the group as a whole. Co-workers make autonomous decisions when working alone, under changing and unpredictable conditions, which the group cannot foresee or plan for.

Our collaborative multi-agent environment provides the application independent framework to meet the infrastructure needs of everyone involved in collaboration. The goal of our collaborative environment is to transform the organization's data into information that can be applied to workflow monitoring with the purpose to improve the performance of business operations. With the multi-agent environment, collaboration goes beyond the simple, serial exchange of information between varied disciplines or domains. It ensures that knowledge can be shared and used at the moments when it can best impact the work.

In our monitoring system, Planning Agent, Information Agent, User Agent and several Searching Agents work autonomously and collaboratively to perform flexible monitoring activities. The collaboration among these agents is supported by the common goal and shared knowledge about securities trading and monitoring. The sequence of such collaboration is presented through an example in Fig. 6. A trade-agreement manager needs to monitor the trade agreement activities in the securities trading process. (1) The manager can send his/her monitoring request

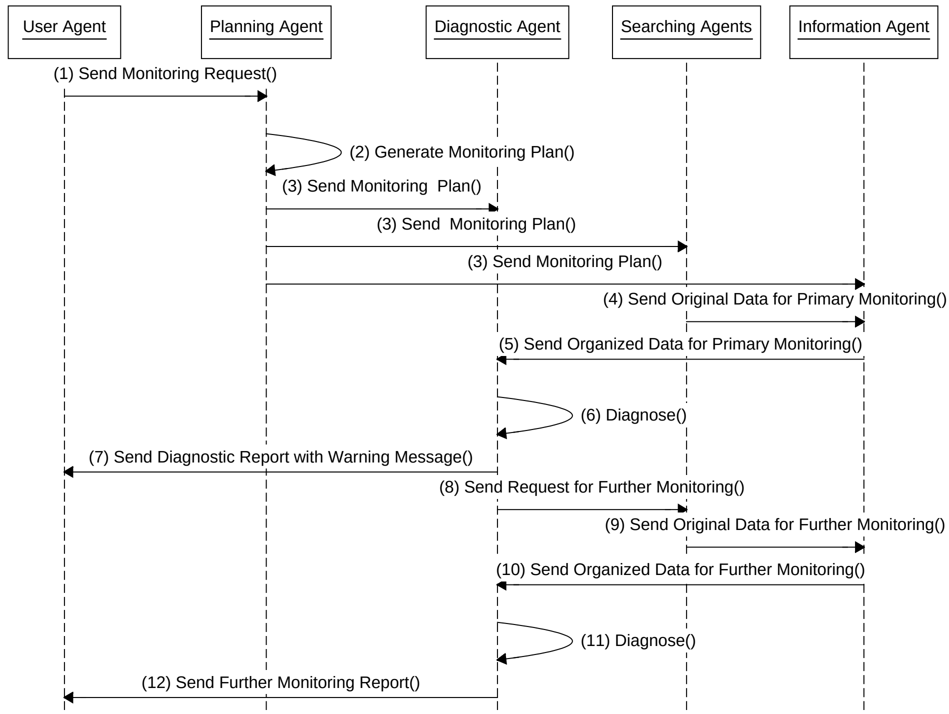

Fig. 6. The sequence diagram of agent collaboration in monitoring activities.

through the User Agent. (2) Based on this request, the Planning Agent generates a customized monitoring plan for this manager, and (3) distributes this plan to other agents. (4) Trade agreement data collected by the Trade-agreement Searching Agent are stored and managed by the Information Agent, and (5) sent to the Diagnostic Agent for analyzing and reasoning. (6) After some analysis, one trade is found to be unmatched in several hours after execution. This problem is reported by the Diagnostic Agent, and (7) sent to the manager through the User Agent. (8) For resolving this problem, the Diagnostic Agent will start its further monitoring process by asking Searching Agents to search trade agreement transmission records for further diagnosing. (9, 10) More information is collected by the Agreement-transmission Searching Agent, and (11) the problem is identified as a transmission error in the communication system. (12) The diagnostic result is also reported to the manager through the User Agent for repair action.

5. Conclusions

With the increased complexity, uncertainty and risks in business operations, workflow monitoring, an important issue in workflow management, has gained growing attention in business process controlling. The major contribution of this paper is the novel architecture of intelligent agents supported flexible and dynamic workflow monitoring. Intelligent agents are designed to provide a number of advanced functionalities for workflow monitoring, such as flexible monitoring plan, proactive monitoring process, and collaboration in monitoring activities. By using customized monitoring plan, agents can execute customized monitoring tasks based on individual monitoring requests from users. With the proactive property of intelligent agents, the monitoring process can be adjusted based on the information from monitoring activities. Through the deployment of a society of agents, complex monitoring tasks can be decomposed and performed in an autonomous and collaborative way. However, the flexibility of our monitoring approach is limited to the monitoring skeletons and monitoring options pre-defined in the system. The future work is to seek more flexibility and adaptability in workflow monitoring for increasingly dynamic business process management.

References

[1] D. Baker, D. Georgakopoulos, H. Schuster, A. Cassandra, A. Cichocki, Providing customized process and situation awareness in the collaboration management infrastructure, Proceedings of the Fourth IECIS International Conference on Cooperative Information Systems, 1999 pp. 79–91.

[2] E. Charniak, R.P. Goldman, A Bayesian model of plan recognition, Artificial Intelligence 64 (1) (1993) 53–79.

[3] Q. Chen, U. Dayal, M. Hsu, M. Griss, Dynamic-agents, Workflow and XML for E-Commerce Automation, Lecture Notes in Computer Science, 2000 pp. 314–323.

[4] P.W.H. Chung, L. Cheung, J. Stader, P. Jarvis, J. Moore, A. Macintosh, Knowledge-based process management—an approach to handling adaptive workflow, Knowledge-Based Systems 16 (3) (2003) 49–160.

[5] R.J. Glushko, J.M. Tenenbaum, B. Meltzer, An XML framework for agent-based e-commerce, Communications of the ACM 42 (3) (1999), 106–114.

[6] R.P. Goldman, C.W. Geib, C.A. Miller, A new model of plan recognition, Proceeding of 1999 Conference on Uncertainty in Artificial Intelligence, 1999.

[7] S. Guerlain, P. Bullemer, User-initiated notification: a concept for aiding the monitoring activities of process control operators, Proceedings of the Human Factors and Ergonomics Society 40th Annual Meeting, 1996, pp. 283–287.

[8] A. Guerra, Exception Management: The Safety Net You've Been Looking For? Wall Street & Technology Online, Sep 4, 2002, URL: http://www.wallstreetandtech.com.

[9] N.R. Jennings, P. Faratin, T.J. Norman, P. O'Brien, B. Odgers, Autonomous agents for business process management, Internal Journal of Applied Artificial Intelligence 14 (2) (2000) 145–189.

[10] M. Klein, Coordination science: challenges and directions, Coordination Technology for Collaborative Applications, Springer, Berlin, 1998, pp. 161–176.

[11] K. Liu, L. Sun, A. Dix, M. Narasipuram, Norm based agency for designing collaborative information systems, Information Systems Journal 11 (2001) 229–247.

[12] P.D. O'Brien, W.E. Wiegand, Agent based process management: applying intelligent agents to workflow, The Knowledge Engineering Review 13 (2) (1998) 1–14.

[13] R. Parasuraman, M. Mouloua, R. Molloy, Effects of adaptive task allocation on monitoring of automated systems, Human Factors 38 (4) (1996) 665–679.

[14] A. Puerta, J. Eisenstein, Towards a general computational framework for model-based interface development systems, Knowledge-Based Systems 12 (8) (1999) 433–442.

[15] M. Simmons, Securities Operations: A Guide to Trade and Position Management, Wiley, New York, 2001.

[16] P. Szekely, C.G. Thomas, M.T. Maybury, Introduction to special issue on intelligent user interfaces (IUI 99), Knowledge-based Systems 12 (8) (1999) 401–402.

[17] H. Wang, J. Mylopoulos, S. Liao, Intelligent agents and financial risk monitoring systems, Communications of the ACM 45 (3) (2002) 83–88.

[18] H. Wang, C. Wang, Intelligent agents in the nuclear industry, IEEE Computer 30 (11) (1997) 28–34.

[19] M. Wang, H. Wang, D. Xu, K.K. Wan, D. Vogel, A web-service agent-based decision support system for securities exception management, Expert Systems with Applications 27 (3) (2004) 439–450.

[20] M. Wang, H. Wang, Agents and Web-services Supported Business Exception Management, Proceedings of 8th Pacific Rim International Conference on Artificial Intelligence, Auckland, New Zealand, August 2004, Lecture Notes in Artificial Intelligence, Springer.

[21] M. Wang, H. Wang, K.K. Wan, D. Xu, Knowledge-based exception handling in securities transactions, Proceeding of Hawaii International Conference on System Science (HICSS-37), Hawaii, USA, 2004.

[22] M. Wang, H. Wang, Intelligent agent supported flexible workflow monitoring system, Proceeding of the14th International Conference on Advanced Information Systems Engineering (CAiSE'02), Toronto, Canada, May 2002. Lecture Notes in Computer Science, vol. 2348, 2002, pp. 787–791.

[23] M. Wooldridge, An Introduction to Multiagent Systems, Wiley, England, 2002.

[24] M. Wooldridge, N. Jennings, Intelligent agents: theory and practice, The Knowledge Engineering Review 10 (2) (1995) 115–152.

[25] Workflow Management Coalition, Audit Data Specification, Draft 1.1a, Document Number WFMC-TC-1015, Winchester, 1998.

[26] Y. Yan, Z. Maamar, W. Shen, Integration of workflow and agent technology for business process management, The Sixth International Conference on Computer Supported Cooperative Work in Design, 2001, pp. 420–426.

[27] H. Zhuge, J. Chen, Y. Feng, X. Shi, A federation–agent–workflow simulation framework for virtual organization development, Information & Management 39 (2002) 325–336.