**IDEA GROUP PUBLISHING**
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
**ITB12735**

This paper appears in the book, *Emerging Trends and Challenges in Information Technology Management, Volume 1 and Volume 2* edited by Mehdi Khosrow-Pour © 2006, Idea Group Inc.

# Cooperative and Dialog-Based Multi-Agents in Global Software Delivery Management

Ravi Gorthi, Software Engineering & Technology Labs, Infosys Technologies Ltd., ravi_gorthi@infosys.com

Andie Kurniawan & Nandan Parameswaran
School of Computer Science & Engineering, The University of New South Wales, {andiek; paramesh}@cse.unsw.edu.au

An important characteristic of management, in our view, is the extensive use of dialogs by the project managers in their problem solving and decision making endeavor. High degree of dependency on manual interpretation in dialogs might cause problems or even failures due to the unpredictable nature of management, inconsistency and egoistic behavior of humans, and the drop of efficiency of humans under high-pressure situations. In our work, we have chosen the field of Global Software Delivery Management (GSDM) as the focus of our application. This paper discusses a cooperative and dialog based multi-agent framework to automate a class of management tasks in GSDM. We describe the important characteristics of GSDM, identify certain experience based, less ill-structured tasks that the project and quality manager routinely perform and discuss why and how the concepts and techniques of our multi-agents framework are suited best to automate the execution of these tasks. This work has the potential to contribute to enhanced and consistent productivity in GSDM. One may note that even a 5% enhancement in productivity and consistency will have considerable revenue benefits. We have implemented a prototype of our work using Java for Agent Development (JADE) platform and we argue here the results and benefits from a very relevant case study.

## 1. INTRODUCTION

There has been of late an increased focus on the applicability of the concepts and techniques of multi agents systems to the field of business management. The initial results have been encouraging [8] [10]. An important characteristic of business management, in our view, is the use of dialogs by a community of professionals to solve problems. One of the serious problems confronting the business managers, in their problem solving and decision making endeavor, is the high degree of dependency on human interactions or the high degree of manual interpretation of dialogs by humans. Such a dependency on human manual intervention lends itself to problems that can result from either: (i) the unpredictable, inconsistent egoistic behavior of humans that we witness time to time, (ii) the drop in efficiency of humans under stressful situations, or (iii) the use of less experienced/qualified humans for managing tasks due to lack of sufficient number of adequately skilled human resources.

On an initial analysis, one would find that the concepts and techniques of intelligent, cooperative, dialog based multi-agent systems can address some of the issues of business management mentioned previously. This technology has the potential to offer: (i) agents' behavior can be designed as desired by the dialogs designer and thus behavior can be predicted, (ii) consistent performance even under stressful situations for humans, or (iii) agents might perform automated decision making using their ability to cooperate among themselves.

We are thus encouraged to apply the techniques of multi-agent systems to enhance the degree of automation in problem solving by business managers. We have chosen the field of Global Software Delivery Management (GSDM), which is a part of Global Delivery Model [6] as the focus of application of our ideas for the following important reasons:

- The field of GSDM is an emerging field of management with a prediction to grow [5]
- The field is not highly ill-structured, thus lending itself to the application of multi-agent systems to enhance the degree of automation in software project, quality and delivery management.

GSDM is one of the management techniques used by IT service providers to exploit time and geography as resources and thereby attempt to reduce or optimize the time and cost of software production and maintenance. GSDM consists of geographically distributed project management and software engineering professionals, who use processes and tools in their endeavor to achieve the functional objective of manufacturing (all phases of manufacturing including maintenance) and delivering quality software. GSDM involves intelligent break-down, allocation and management of tasks to achieve the twin goals of manufacturing: (i) quality software components at (ii) reduced / optimized costs. GSDM professionals, like other business management professionals, use dialogs to seek cooperation and service from their co-professionals in order to accomplish their tasks and goals.

We have studied and analyzed the field of GSDM and found that the concepts and techniques of multi-agent systems are quite useful to provide 'intelligent assistance' to the geographically distributed project and quality managers in accomplishing some of the routine but experience based tasks. We have also found that the traditional software development models, such as procedure or object oriented, are not directly suitable to automate these experience based project / quality management tasks because these tasks vary from project to project and / or time to time and hard-coded solutions (tools) to these tasks lead to considerable problems during maintenance. We have discovered, as presented in this paper, that the concepts and techniques of multi-agent systems are in fact better suited.

This paper is organized as follows. In section 2, we provide an outline of the related research work. In section 3, we discuss the results of our research work on the use of intelligent, cooperative and dialog-based multi-agent system to automate certain tasks in GSDM. It also illustrates our work with a case study. In section 4, we describe how we have modeled the intelligent agents using the JADE (Java Agent Development Environment) framework and platform [7]. Section 5 presents the conclusions from this research work and directions for future course of work.

## 2. RELATED RESEARCH WORK

As a promising approach for project management area, intelligent agent-based system supports framework that defines, manages, and

executes management processes through their automation and proactive approaches. Several approaches have been made to provide solutions in this area [3] [8] [9][10].

Jennings et al proposed an agents framework for business process management in [8]. The key feature of their system, known as ADEPT, is that the responsibility for enacting various components of a business process is delegated to a number of autonomous problem solving agents. These agents typically interact and negotiate with other agents in order to coordinate their actions and to buy in the services they require. O'Connor et al [10] investigate the use of agents in the distributed software project management. Agents in this framework provide guidance to human users. Concurrence from an appropriate human being (e.g. authorized staff) is sought as to whether or not to undertake the set of actions. This concept hinders the agents to be highly autonomous due to a large degree of human intervention.

Nienaber [9] proposed a set of agent teams, wherein each agent is given an atomic task, to address different tasks of the management activities. In our research, we use a single agent at each individual staff's desktop and at each server to manage the activities. And, our focus is much larger than that of Nienaber. In addition, our dialog based multi agent framework for management tasks is more appropriate for an environment that is characterized by many types and higher complexity of activities compared to other approaches. By having a generic type of agent in the activities, one can simply append more rules in the knowledge database to support new activities.

Foundation of Intelligent Physical Agent (FIPA) provides a number of interaction protocol guidelines such as contract net, Dutch auction, and English auction protocols. These protocols often provide simple but effective and efficient way to distribute the activities to accomplish a common goal among the agents and have been used in many multi-agents applications. However, Norman [2] argued that the interaction protocols offered by FIPA are inflexible. Our dialogs model that is developed based on the dialogs theory rooted from communication study offers higher flexibility by constructing rules based system for the automated interaction among agents during execution and proactively adjusting the mental states of each agent. Parsons [11] proposed a formal framework of dialogs based on the theory of deliberative reasoning from the philosophy.

## 3. OVERVIEW OF OUR WORK

Given the aforementioned goal of this research work, our initial focus has been, (i) to study, analyze and find out the classes of tasks in Global Software Delivery Management that can be automated using intelligent, cooperative and dialog-based multi-agents, and (ii) to model such intelligent agents using the JADE framework and platform.

In this and the next section, we describe the results of this research work.

### 3.1. Study and Analysis of GSDM

As stated earlier, GSDM involves intelligent break-down, allocation and management of tasks to achieve the twin goals of manufacturing: (i) quality software components at (ii) reduced / optimized costs. From a study and analysis of GSDM, we have observed that there are certain tasks that the GSDM managers routinely perform. Examples of such tasks include:

- *Coordination* of 'Quality Review of Artifacts';
- Dynamic *work allocation / re-allocation* based on requirements of job role and skills of personnel;
- Controlled *distribution* (including timely destruction) of electronic copies of confidential documents, etc.

These tasks have the following characteristics:

- These tasks do not generally require the experience of the GSDM managers.
- These tasks are required to be performed periodically during GSDM (and thus are not one-off tasks).
- The business rules / policies governing these tasks vary from project to project and / or time to time.
- To complete these tasks, effective communication and coordination among GSDM personnel are required most of the time.
- These tasks are not so well structured that they can be automated using the traditional software development models like procedural or object-oriented models.
- Yet, these tasks are not so ill-structured that they cannot be automated using intelligent (knowledge-based) agents.

### 3.2. Dialog Model

From a study and analysis on how the above tasks are manually accomplished in real-world projects, we have come up with the following classification of activities related to these tasks:

- **Type-0 Activities:** These activities are related to accomplishing planning tasks locally (e.g. selecting best possible candidates for review process);
- Type-0 Activities: These activities are related to initiation, coordination and completion of many general planning tasks; these may refer to a knowledge base (e.g. rule base) to make intelligent decisions (e.g. under what conditions to allow distribution or destruction of an electronic copy of a confidential artifact);
- **Type-1 Activities:** These activities are related to the interactions between an agent and a human being in the form of speech-based_(Voice User Interface) and/or, text-based communications through mobile phones and PDAs;
- **Type-2 Activities:** These activities are related to initiation, coordination and completion of many general planning tasks and involves cooperation works among agents; these may refer to a knowledge base (e.g. rule base) to make intelligent decisions (e.g. monitoring the progress of a review process).
- Type-2 Activities: These activities are related to accomplishing specialist jobs (e.g. how to permanently destroy an electronic copy of an artifact from a UNIX based server or a Windows based server or a PDA).

We represent a dialog *D* by the following set of attributes

- *I*, the dialog identification label;
- *P*, the set of dialog participants;
- *C*, the set of conditions that trigger the activation of dialog *D*;
- *M*, the set of dialog modules monitored by the dialog *D*.

We restrict ourselves to dialogs carried out by two agents and can be one or more of the following types of dialogs: (i) **Instructional Dialogs,** used by an agent to proactively delegate responsibilities to other appropriate agents, (ii) **Inquiry Dialogs,** used by an agent to seek information from other agents, (iii) **Informational Dialogs,** used by an agent to communicate the response of an allocation request and/or the status of a sub tasks performed by other agents, and (iv) **Negotiation Dialogs:** used by an agent to negotiate in an attempt to find a solution to a common problem cooperatively.

In our work, these dialogs will be structured into dialog recipes. In brief, a dialog recipe is a complex set of dialog modules (*M*) that is executed in full/partial order and comprises performative and communicative modules. It contains a set of nodes (*M*) and their temporal relationship. There are two major types of the dialog modules: (i) control nodes, contain nodes that specify the control flow of *D*, and (ii) action nodes, contain nodes that specify the action to be executed.

### 3.3. Case Study: Quality Review Process

One of the important tasks of a Quality manager of GSDM is to periodically conduct quality reviews of various artifacts like analysis documents, design documents, code, test-case documents, etc. By an analysis of this process, we have found out that this task can be

automated using intelligent, cooperative and dialog-based multi-agents. In this sub-section, we discuss this case study.

The task of coordinating the quality review process consists of (i) Finalizing the list of reviewers, finalize the list of three reviewers who can and have accepted to perform the review (ii) Forwarding electronic copies, the electronic copies of the artifacts to be reviewed (along with reference documents) are forwarded to all the reviewers (iii) Coordinating the reviews, the review process activities need to be coordinated (iv) Coordinating the feedback resolution, Inform the owners (software engineers) of the artifacts about the review feedback and (v) Closing the review process, formally certify that the artifacts are reviewed and issues if any are resolved. This paper will cover the first stage of review process in depth, 'Reviewers Selection'.

*3.3.1. Finalizing the list of reviewers*

The aim of this activity is to finalize the list of three reviewers who can and have accepted to perform the review of the given artifacts; the selection of these reviewers depends upon what type of artifacts are reviewed (e.g. design documents or test-case documents) and the availability of the skilled reviewers; the selection also takes into account factors like which reviewer was least recently used; this activity usually takes around 3 working days to finalize the list of 3 reviewers and thus it is initiated a week in advance of the scheduled period (dates) of review process.

*3.3.2. Reviewers Selection Dialog*

This dialog involves 3 types of dialog module types mentioned in 3.2.: inquiry (e.g. sending availability request to other agents), informational (e.g. sending response upon receiving review request), and negotiation (e.g. negotiating upon conflict of interest).

Figure 1 shows the structure of the dialog recipe mentioned previously. Five different nodes can be observed from the diagram. Each type of nodes has different property that specifies the execution policy of actions. The nodes shown by normal lines specify the modules that need to be executed and then the agent proceeds to the next stage. The parallel nodes (e.g. double lines boxes) specify the set of actions which need to be executed in parallel by the agent (e.g. sending questions to find out similarity measure). Then, the nodes shown by dashed lines specify a set of actions that may/may not be executed by the agent depending upon the situations. More than one action can be selected in the optional nodes. Start and end nodes are special nodes that are designed to indicate that the recipe start and end consecutively.

Similar dialogs are used in the other review process stages.

## 4. MULTI AGENTS FRAMEWORK

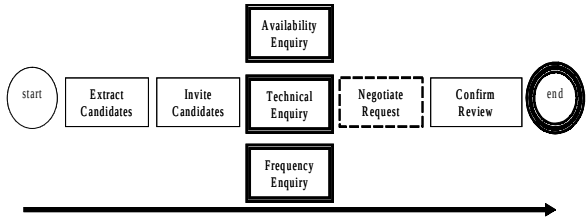
We present an overview of the design and implementation of our intelligent agents in quality review process coordination. First, we will briefly describe some of the important modules in the agent architecture. We then look at how we model the GSDM environment to facilitate the simulation. Then, we look at the architectural view of the simulated review process coordination and the dialog structure during the execution. We also present some screenshots from our system and we conclude this section with performance analysis of our framework.

In this section, we first outline how we can use this system in the real world situation by mimicing the actual GSDM environment. Then, we discuss the organizations of agents in the quality review process by discussshowing the top level architecture of our multi agent system. We then briefly look at the components of the system and the implementation details of dialog recipes structure in quality review process. Some screenshots will be displayed and we conclude this section with performance analysis of our framework.

### 4.1 Organizations of Agents in Quality Review Process

Currently, the system comprises the following type entities: (i) one static agent on each desktop of the team members at each location and

*Figure 1: Reviewers selection dialog diagram*




(ii) one mobile device agent for the team members at each location. In our example, we assume that there are 10 staffs in the organization. These staffs are assumed to be distributed geographically, own desktop and mobile device with MIDP enabled and interconnected through the Internet. These assumptions do not cause any deficiency as our agents are generic and can be deployed for different staffs with different roles easily and thus the simulation captures their interaction adequately.

### 4.2. Implementation Model

To carry out performance evaluation of our multi agents framework in GSDM, we have developed a prototype application of dialog-based intelligent assistants in quality review process coordination within GSDM field. We chose JADE as the agent platform since the goal of JADE is to simplify the development of multi agent system development while ensuring standard compliance through a comprehensive set of system services in compliance with FIPA standard [7]. In addition, LEAP and JADEX libraries are used to add mobile device agents feature and BDI framework consecutively to the agent framework.

Mobile device agents are used to facilitate interaction between agents and the GSDM personnel considering their flexibility nature and virtually low cost. In this paper, the mobile devices are used for monitoring purposes only.

### 4.3. Components of Our Multi Agents Framework

Our system adopts Jadex agent model by incorporating beliefs, goals and plans that can be modified during runtime execution. The architecture of our system is depicted in Figure 2.

The agent reacts to messages from the environment and internal events. Then, based upon these information, the agent selects and executes plans, which in our system can be dialog plans and/or performative plans.

### 4.4. Experimental Result

We look at the comparative analysis between the dialog recipe system and simple ACL Messages exchange approach on examining the percentage of reviews being escalated to the project leader in case the requirement of review (e.g. 3 reviewers should be confirmed) is not met. The best case scenario is all of the reviews can be finalized without project leaders' intervention. From the graph in Figure 3, the performance of both approaches is similar when the number of available candidates is lower than the required number of reviewers. After that, the performance of dialog recipe framework overruns the other approach. This enhancement occurs due to agents' ability to negotiate on the required factors as mentioned in the previous subsection; availability, ability and frequency. In the case where one or more requirements are not met, project leader agent will try their best to force the reviewers' agent to accept the request by way of negotiation.

### 4.5. Merits of our Framework

Let us look at the impacts of applying multi agent paradigm in the quality review process coordination. Benefits of introducing dialog-based agents in 'Quality Review Process' include:

*Figure 2: System Architecture*

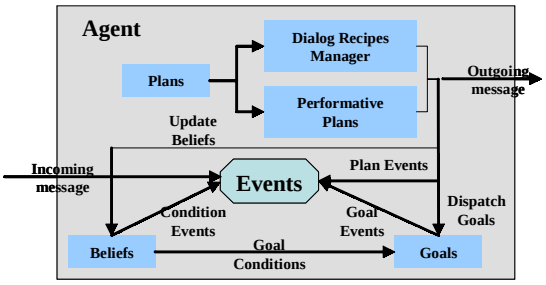
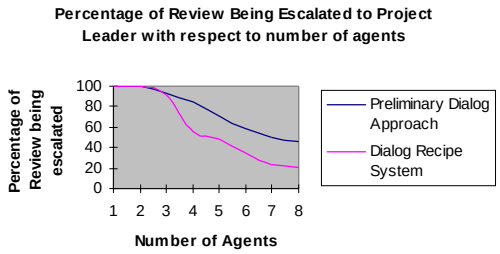



*Figure 3. Probability of review being escalated to PL*




- Eliminating considerable amount of time spent on mundane, manual activities of selecting and finalizing a set of reviewers;
- Reducing the amount of time to manage the review process delays using easy to change rules;
- By ensuring that the reviewers are able to access the electronic copies of artifacts through controlled circulation of the copies, considerable time is saved

## 5. CONCLUSIONS

In this paper, we presented the results of applying the concepts and techniques of intelligent, cooperative and dialog-based multi-agent systems to automate certain tasks in Global Software Delivery Management (GSDM). We described the important characteristics of GSDM, identified certain experience based, less ill-structured tasks that the project and quality managers routinely perform and discussed why and how the concepts and techniques of multi-agent systems are better suited to automate the execution of these tasks and thus have the potential to contribute to enhanced and consistent productivity in GSDM.

One may note here that even a 5% enhancement in productivity and consistency will have considerable revenue benefits. We have modeled our ideas using the JADE framework / platform and presented here the results and benefits from a case study from the field of GSDM.

While our future work are aimed at extending the features of intelligent assistant to be able to carry out more project management task in GSDM and evaluating the benefits of applying dialog-based multi agent system in GSDM, on the theoretical side, we plan to extend the dialog recipes and formalize it.

## 6. REFERENCES

[1] Au, S., Liang. J, and Paramesh, N. *A Recipe Structure for Plan Execution in Dynamic Worlds*. CoopIS, 1998.

[2] Carbogim, D. V., Norman, T. J., Krabbe, E. C. W., and Walton, D., *Argumentation and Multi-Agent System*, In Reed, C., and Norman, T. J. (Eds), Argumentation Machines: New Frontiers in Argument and Computation (Vol. 9, pp. 15-52), Dordrecht, The Netherlands: Kluwer Academic Publishers.

[3] Drucker, P., *Management Challenges for the 21st century*, Harper Business, 1999.

[4] Harris, J. G., and Davenport, T. H., *Automated Decision Making Comes of Age*, Research Report, Accenture: Institute for High Performance Business, 2005.

[5] Caldwell, B. M., et al., *Continued Growth Forecast for IT Outsourcing segments*, Gartner Research Report # G00124277, October, 2004, http://www.gartner.com.

[6] *Global Delivery Model*, Infosys Technologies Limited, http://www.infosys.com, 2006.

[7] JADE Programmer's Guide, JADE Board, January 2005.

[8] Jennings, N. R., Faratin, P., Johnson, M. J., Norman, T. J., O'Brien, P., and Wiegand, M. E., *Agent Based Business Process Management*, International Journal of Artificial Intelligence, 2002.

[9] Nienaber, R., and Barnard, A., *Software Quality Management supported by Software Agent Technology*, Information Science + Information Technology Education Joint Conference, 2005.

[10] O'Connor, R., and Jenkins, J., *Using Agents for Distributed Software Project Management*, International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises, 1999.

[11] Parsons, S., Hitchcock, P., and McBurney, P., *The Eightfold Way of Deliberation Dialogue*, International Journal of Intelligent System, 2004.