International Journal of Mechatronics, Electrical and Computer Technology

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

Modeling an Agent Mediated Supply Chain Management System Using MAS-CommonKADS Methodology

Mohammad Sabri[1] and Marjan Fatemi Garakani[2]

[1] Department of Electrical and Computer Engineering, Islamic Azad University, Mahshahr Branch, Mahshahr, Iran
[2] Departments of Computer Engineering, Islamic Azad University, Damavand Branch, Damavand, Iran

*Corresponding author's E-mail: *M.sabri@mahshariau.ac.ir*

Abstract

Supply chain is a worldwide network of suppliers, factories, warehouses, distribution centers, and retailers through which raw materials are acquired, transformed, and delivered to customers. In recent years, new software architecture for managing the supply chain at the tactical and operational levels has emerged. It views the supply chain as composed of a set of intelligent software agents, each responsible for one or more activities in the supply chain and each interacting with other agents in the planning and execution of their responsibilities. This paper uses the MAS–CommonKADS methodology in modeling a supply chain management system. We develop each model included in this methodology, illustrating the development of both the coordination and expertise models. We incorporate UML activity diagrams in the task model and use sequence diagrams to model communication between agents. The methodology leads in defining five different classes of agents

Keywords: MAS-CommonKADS, Supply Chain Management, Methodology, Multi Agent System

1. Introduction

MAS-CommonKADS extends CommonKADS, for multi-agent systems modeling, adding techniques from object oriented (OO) methodologies such as Object Modeling Technique (OMT), Object Oriented Software Engineering (OOSE) and Responsibility

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

Driving Design (RDD) and from protocol engineering for describing the agent protocols, such as Specification and Description Language (SDL) and Message Sequence Charts (MSC96) [1]. MAS-CommomKADS [2] comprises seven models that cover the main aspects in the development of multi-agent systems. The agent model specifies agent characteristics such as reasoning capabilities, sensors/effectors, services, agent groups and hierarchies. The task model describes the task that the agents can carry out, for instance goals, decomposition, problem-solving methods, etc.

The expertise model defines the knowledge needed by the agent to achieve their goals. The organization model describes the social organization of the agent society. The coordination model illustrates the conversation between agents. The communication model details the human-software agent interactions. The design model includes, in addition to the typical action of the design phase, the design of relevant aspects of the agent network, selecting the most suitable agent architecture and the agent development platform. Supply Chain Management is the most effective approach to optimize working capital levels, streamline accounts receivable processes, and eliminate excess costs linked to payments. Analysts estimate that such efforts can improve working capital levels by 25% [3].

Today, the best companies in a broad range of industries are implementing financial supply chain management solutions to improve business performance and free cash resources for growth and innovation. A Supply Chain is a network of suppliers, factories, warehouses, distribution centers and retailers, through which raw materials are acquired, transformed, produced and delivered to the customer [4]. In addition Supply Chain Management is about managing the physical flow of product and related flows of information from purchasing through production, distribution and Delivery of the finished product to the customer. This requires thinking beyond the established boundaries, strengthening the linkages between the supply chain functions and finding ways to pull them together. The result is an organization that provides a better service at a lower cost. Many managers now realize that actions taken by one members of the chain can influence the profitability of all others in the chain. Two firms are increasingly thinking in terms of competing as part of a supply chain against other supply chains, rather than as a single firm against other individual firms. Also, as firms successfully streamline their own operations, the next opportunity for improvement is through better coordination with their suppliers and customers. The costs of poor coordination can be extremely high [5].

To the best of our knowledge an integrated, agent based methodology analysis and design approach to the supply chain procurement, production and customer order bidding problem has not been addressed in the literature thus far. Much of the literature in production/inventory planning has looked at the procurement and production aspects of the problem, though oft en in isolation [6]. Thomas and Griffin [6] provide an excellent review of these models. Two not able exceptions are the work by Bassok and Akella [7] and Sun and Sadeh [8]. Bassok and Akela develop a single period model where both procurement and production costs are considered. They assume a single-machine production scenario with a single critical raw material and multiple products with stochastic demand. The authors demonstrate the benefit, of jointly optimizing the production and procurement decisions. Sun and Sadeh consider the integrated procurement and production problem of a manufacturer in a single period during which multiple stochastic customer bids need to be satisfied.

The customer bids and procurement offers vary in terms of product type, due date and tardiness penalties. The customer bids are satisfied by selecting suitable component procurement offers from a range of available options. The objective is to select procurement bids and devise a production schedule so as to minimize procurement and tardiness costs. The authors present a number of heuristics derived from a set of dominance rules to help prune the search space to arrive at a selection of procurement bid combinations. While the Production/planning literature has largely ignored the bidding and bid selection aspects of the problem, the electronic commerce community has primarily focused on this aspect but oft en ignored capacity and temporal constraints. Babaioff and Nissan [9] propose bidding protocol s that results in efficient allocation of goods among the supply chain partners in a linear supply chain. They organize auctions

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

in terms of a series of production markets and derive a double auction mechanism that has the attractive feature of being incentive compatible (IC), individually rational (IR), and results in a balanced budget (BB). Babaioff and Walsh [10] further ext end the work of Babaioff and Nissan to a broader class of supply chain topologies. These protocols retain the properties of IC, IR, and BB, while still maintaining high allocation efficiencies. Walsh and Wellman [11] have proposed a family of decentralized protocol s based on the task dependency network model, to negotiate supply contracts. All these models operate under a single period and assume only single unit transact ions thus ignoring capacity and temporal constraints.

Other related work includes the work by Sadeh [12] on MASCOT, an agent based supply chain decision support tool that supports finite capacity models and provides available-to-promise and profitable-to-promise functionalities. Zeng and Sycara [13] develop a real time supply chain formation model with the view of studying inventory decisions by taking into account multiple lead-times versus cost opt ions. A significant work, though not directly related to goods exchange that takes into consideration capacity and temporal constraints in the scheduling of tasks is the work on MAGNET [14].

MAGNET provides a framework were agents negotiate the coordination of tasks constrained by temporal and precedence relationships. In the computational world, roles of individual entities in a supply chain can be implemented as distinct agents. Correspondingly, a SCMS transforms to a MAS, in which functional agents cooperate with each other in order to implement system functionality [15]. The functions and procedures of a company in the real market are complicated and include information collection, policy making and actions. Therefore, it is impossible to describe software agent behaviors for an uncertain e-commerce environment such as supply chain management in the traditional single threaded model. To solve the problem, we introduce the multi-agent model which agents negotiate with each other to manage the supply chain using the MAS-CommonKADS methodology.

This paper is organized as follows. Applying of methodology includes conceptualization and design is describe in section 2. Implementation of agents is given in section 3. Section 4 presents our conclusions.

2. Applying of methodology

Position MAS-CommonKADS extends CommonKADS [16], for multi-agent systems (MAS) modeling, adding techniques from object oriented (OO) methodologies such as Object Modeling Technique (OMT) [17], Object Oriented Software Engineering (OOSE) [18] and Responsibility Driving Design (RDD) [19] and from protocol engineering for describing the agent protocols, such as Specification and Description Language (SDL) and Message Sequence Charts (MSC96) [1]). The methodology defines the following models:

– Agent model (AM): specifies the agent characteristics: reasoning capabilities, skills (sensors/effectors), services, agent groups and hierarchies (both modeled in the organization model).

– Task model (TM): describes the tasks that the agents can carry out: goals, decompositions, ingredients and problem-solving methods, etc.

– Expertise model (EM): describes the knowledge needed by the agents to achieve their goals.

– Organization model (OM): describes the organization into which the MAS are going to be introduced and the social organization of the agent society.

– Coordination model (CoM): describes the conversations between agents: their interactions, protocols and required capabilities.

– Communication model (CM): details the human-software agent interactions, and the human factors for developing these user interfaces.

– Design model (DM): collects the previous models and consists of three sub models: network design for designing the relevant aspects of the agent network infrastructure (required network, knowledge and telematic facilities); agent design for dividing or composing the agents of the analysis, according to pragmatic criteria and selecting the most suitable agent architecture for each agent; and platform design for selecting the agent development platform for each agent architecture.

3. Conceptualization

In this section we will describe the modeled system. The first step in this methodology is the conceptualization phase after which an elicitation task will be carried out to obtain a general description of the problem by following a user-centered approach based on use cases. In this approach, an actor represents a role played by a person, a piece of hardware or another system that interacts with our system. A use case corresponds to a description of the sequence of actions needed to produce an observable result useful for an actor. Table 1 defines the actors and their use cases modeled in our experiment.

Table 1- Actors and Use Cases

Actor	Description	Use case
Repository Agent	A software agent which manages the repository	Request raw materials Provide information about the repository Save materials in repository
Supplier Agent	A human agent that supplies the materials	List recommendations Determine price Provide raw materials
Buyer Agent	A software agent which negotiates about the price and the quantity of needed raw materials	Determine recommendations Negotiate about price and quantity Negotiate how to transmit materials Buy materials

International Journal of Mechatronics, Electrical and Computer Technology

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

Transmission Agent	A human agent who transmits raw materials from supplier to company	Transmit materials to company Deliver materials to repository
Coordinator Agent	A software agent that coordinates other agents	Coordination Find appropriate agent

The outcome of this phase is a description of the different actors and their use cases. Table 2 describes a sample use case diagram.

Table 2 – Sample Use Case

Use Case Negotiate Price/Qty **Summary:** Buyer agent negotiates supplier about the price and the quantity of needed materials. It offers a base price and increases it gradually until a pre-defined threshold or till an agreement takes place. Then, it informs the coordinator agent about the agreed price and quantity of materials. **Actors:** Coordinator - Supplier **Precondition:** A request received from repository agent through the coordinator **Exceptions:** Find no supplier providing demanded materials Insufficient materials found.

Fig. 1. shows use case diagram for conceptualization phase which specifies Negotiate Price/Qty use case diagram.

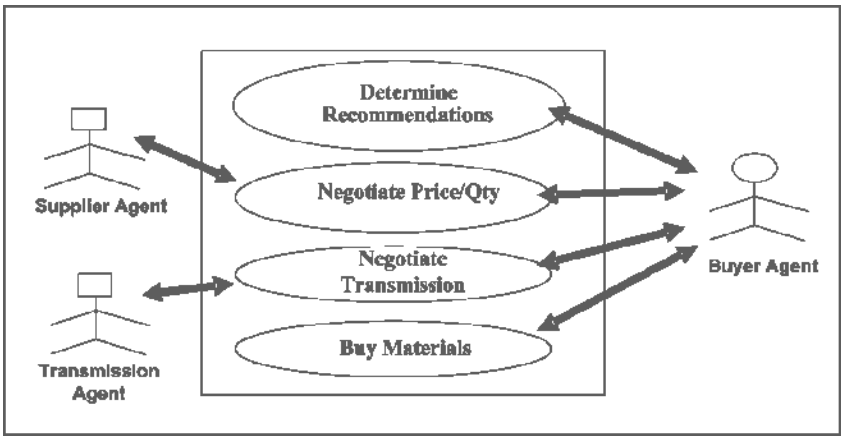

Fig.1: Negotiate Price/Qty use case diagram

4. Analysis

Central to the methodology is the agent model, which specifies the characteristics of an agent, and plays the role of a reference point for the other models. An agent is defined as any entity – human or software – capable of carrying out an activity. The identification of agents was based on the use cases diagrams generated in the conceptualization. Such identification could be augmented in the task model. The agents identified in the agent model are:

Supplier agent: An external software agent who interact with internal agents in company for agreement about price, quality, quantity and who to transmit raw material from supplier to company.

Repository agent: An internal software agent who Manage repository and inform buyer agent about the needed materials.

Buyer agent: An internal software agent who Look for suppliers that supply the needed materials and negotiate with them.

Transmitter agent: A human agent who responsible for transmitting raw material from supplier to company.

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

Coordinator agent: An internal software agent who ensure availability and passing messages between agents.The methodology defines textual templates for each constituent in order to describe it. For instance, Table 3 presents the template for the buyer Agent.

Table 3 – Buyer Agent Textual Template

Agent *Buyer Agent*

Name: Buyer-Agent

Type: Software Agent

Role: price/qty negotiator

Location: Inside agent society

Description: This agent negotiates about the price and the quantity of needed materials.

Objective: It provides a list of suppliers as well as their materials and prices to help the company buy its needed materials.

Exceptions: Find no supplier providing demanded materials
Insufficient materials found.

Input Parameter: Repository information

Output Parameter: possible suppliers for the company

Services: provide information related to the demanded materials

Expertise: The buyer agent gets information about the available materials in the repository using the coordinator agents. It also maintains suppliers' profiles in order to find relevant suppliers easier.

Communication: Supplier Agent

Coordination: Coordinator

Let us now turn to the task model that describes the tasks that the agents can carry out. Since MAS-CommonKADS does not include a graphic structure for modeling tasks, we use UML activity diagrams to represent the activity flows and the textual template to describe the task Figure 2 shows the activity diagram for the Buyer agent and table 4 demonstrates the textual template of a sample task.

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

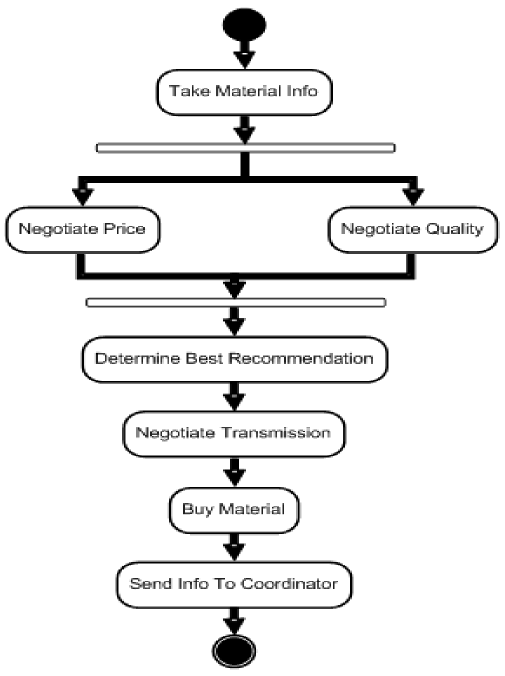

Fig.2. Buyer Agent Activity Diagram

Table 4 – Definition of Take Material Info Task

Task:
Take the Repository Info.
Objective:
Take the needed materials information from repository agent through the coordinator.
Description:
When the repository agent demands some raw materials, it sends it request to the coordinator agent. The coordinator, in turn, will forward it to the buyer agent and then, the buyer agent starts the negotiations.
Ingredients:
Repository information
Constraints:
None
Exceptions:
None

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

This model aims to specify the structural relationships between human and/or software agents, and the relationships with the environment. In the organization model, we show the static or structural relationships between the agents. Following [1], we use a graphical notation based on OMT [17] to express these relationships. Figure 3 illustrates the class agent diagram for our agent system.

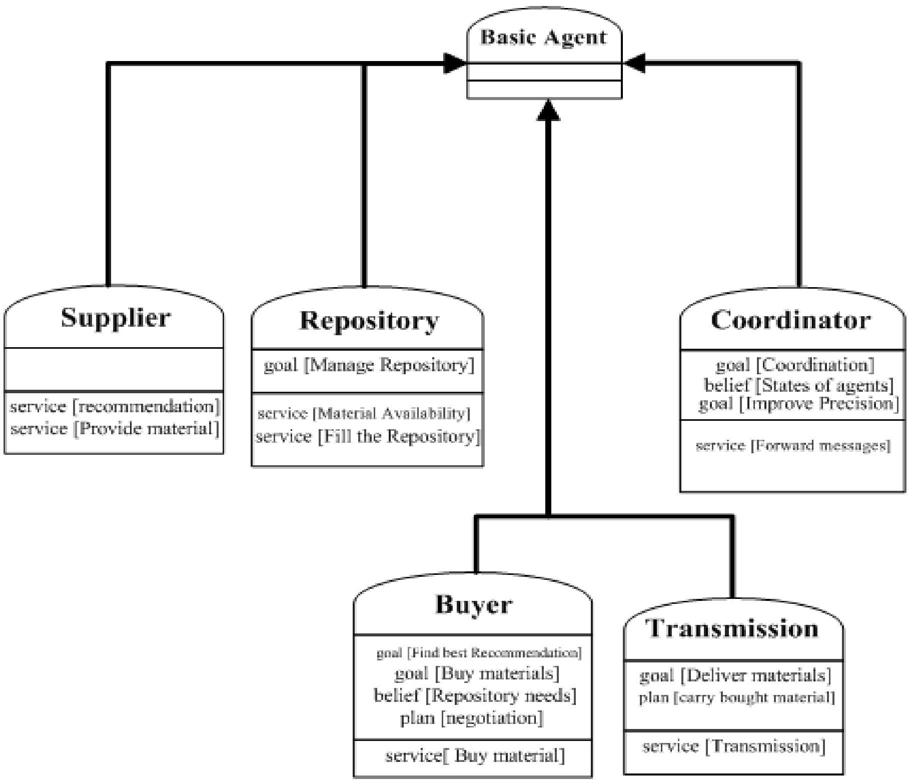

Fig.3. Organization model

Unlike the organization model, the coordination model shows the dynamic relationships between the agents. In this model we begin with the identification of the conversations between agents, where use cases play again an important role. At this level, every conversation consists of just one single interaction and the possible answer, which are described by means of templates as illustrated in Table 5.

International Journal of Mechatronics, Electrical and Computer Technology

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

TABLE 5 – CONVERSATION DETERMINEPRICE TEXTUAL TEMPLATE (COORDINATION MODEL)

Conversation *Determine Price*
Type: Performance
Objective: Get materials prices from supplier
Agents: Buyer, Supplier
Beginner: Buyer
Services: Determine the prices of materials
Description:
The buyer agent determines the prices of required materials by asking from the supplier. It first finds some proper suppliers which may have demanded material. Then, it asks them for the prices.
Precondition: The information about the repository is presented.
Postcondition: The buyer knows the list of available suppliers and their issued prices.
Ending Condition: no suppliers found– no agreements on prices

Next, we model the data exchanged in each interaction by specifying speech acts and the sync synchronization hronization type. We collect all this information in the form of sequence diagrams and textual templates as shown in Figure 3. We also use templates similar to those of the coordination model, but taking into consideration human factors such as facilities for understanding the recommendations given by the system.

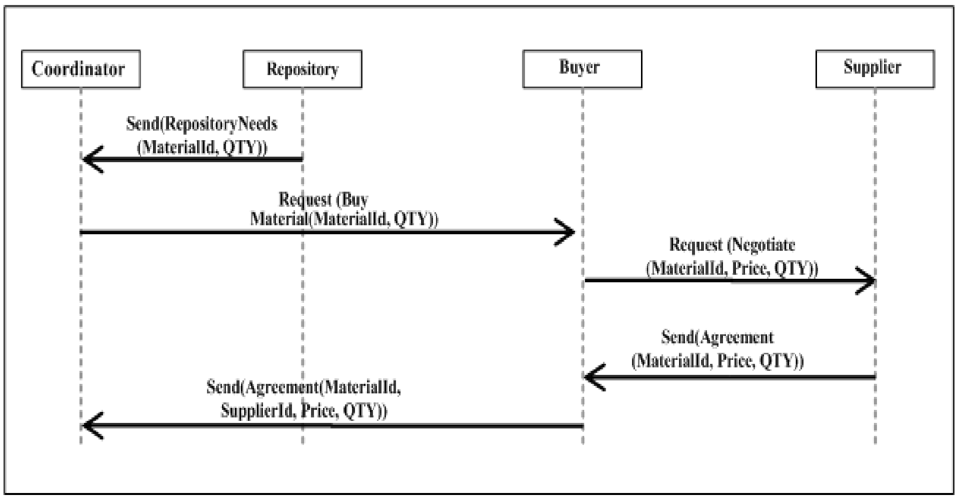
Coordinator
Repository
Buyer
Supplier
Send(RepositoryNeeds (MaterialId, QTY))
Request (Buy Material(MaterialId, QTY))
Request (Negotiate (MaterialId, Price, QTY))
Send(Agreement (MaterialId, Price, QTY))
Send(Agreement(MaterialId, SupplierId, Price, QTY))

Fig.4. Sequence Diagram of a sample Conversation

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

The expertise model is split into the development of the application knowledge and the definition of the problem solving method. In order to develop the application knowledge, we determine the domain knowledge, which defines the ontology's and models of the domain; the task knowledge, which specifies the knowledge needed by a task to reach its goals; and the inference knowledge, which represents the inference steps needed to solve a task. In our case study, the domain knowledge consisted of a set of concept definitions. The task knowledge is represented in Table 6.

Table 6 – Generic tasks of a SCM system

Agent	Generic Task	Knowledge
Supplier	None	None
Repository	Manage repository Inform about the needed materials	Available catalogue
Buyer	Look for suppliers that supply the needed materials and negotiate with them	Available suppliers, Previous suppliers Suppliers' profiles Previous negotiations
Transmission	None	Bought materials and their suppliers
Coordinator	Availability Forwarding messages between agents	Available agents Agents profiles Agents services

We use an inference diagram to model the inference domain. Figure 5 illustrates the case of one of our task, the supplier selection task. In this diagram, the boxes represent information sources, the ovals illustrate the inferences made by the Supplier selection agent and arrows indicate information flows between the information sources and the inferences.

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

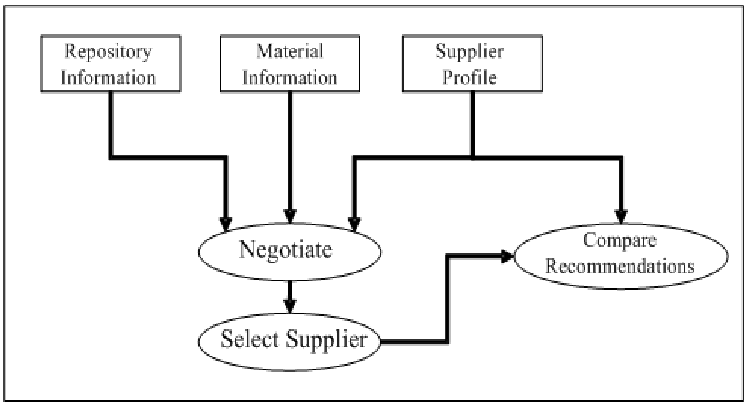

Fig.5. Supplier Selection Inference Diagram

In the definition of the problem solving method, we use ASSESSMENT as the methodology to generate schemes for organizing the knowledge. This methodology includes the representation of the relevant knowledge (i.e. the knowledge that generates special actions in the system, for instance special discounts for older suppliers), and its definition in a representation language. We use the language associated to JESS as the representation language, as illustrated in Table 7.

Table 7 – Knowledge representation using JESS syntax

Supplier Concept	**Supplier Discount Rules**
(deftemplate supplier	**(**defrule supplierdiscount
(slot Id)	(profile (CooperationRecord ?x))
(slot name)	(test (gt ?x 7))
(slot address)	=>
(slot email)	(recommendation (Cost ?cost))
(slot IdType)	(recommendation (supplier ?discount))
(slot CooperationRecord)	(bind ?rate(-?cost (* ?cost ?discount)))
(slot memberno))	(recommendation (topay ?y))
	(recommendation (topay (-?y ?rate))))

Table VII illustrates the construction of the Supplier concept and the Supplier Discount Rule using JESS as a representation language. The Supplier concept (left-hand side) involves the creation of slots (attributes) that determine the concept structure. Furthermore, the Supplier Discount rule (right-hand side) shows the construction of a simple rule, in which the CooperationRecord slot of a supplier profile is checked in order to determine if the corporation's membership period has reached seven years, the discount factor is taken into account. If this is true, it will determine the value of the supplier discount from the Recommendation concept. The new value to be paid is the result of the subtraction of the supplier discount from the (full) price of a material.

5. Design

The design model consists of the design of the agent network, the agent and the platform. The agent network consists of a set of agents that maintain the network, knowledge and coordination facilities. As part of the process of agent design, we determine the most suitable architecture for each agent by decomposing each agent in modules for user communication (from the communication model), agent communication (from the coordination model), deliberation and reaction (from the agent, expertise and task models).

6. Implementation

The implementation and test phases are not part of the methodology, since they depend on the employed platform. The first step in the implementation consisted in the construction of the user interface, based on an interface flow diagram generated from the communication model, as shown in Figure 7. Each interface element activates a process developed by either an agent or a specific class. We employed Java as the programming language for the implementation of the agent and classes. To include the expertise, we use JESS, the Java Expert System Shell tool [16], which consists of a set of Java classes in order to building and operating knowledge databases, following an object-oriented approach.

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

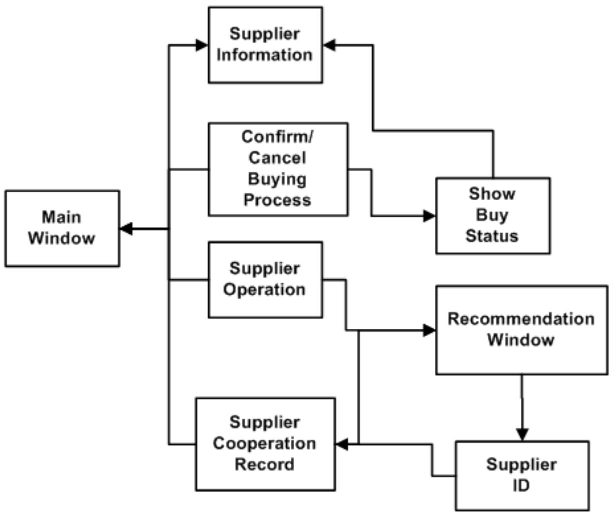

Fig.6. Supply Chain Management system interface flow diagram

Conclusion

We have analyzed the supply chain management using MAS-CommonKADS methodology. We believe that the methodology is particularly appropriate for generic, component-based systems for our application that can be used in a variety of operating environment and computing platform. An interesting outcome of this analysis is the need for agent-oriented methodologies (and indeed those elements describing internal design of agents) to be clear about the seven models of this methodology that apply to supply chain management. We believe we have contributed in several ways to the goal of constructing models and tools enabling multi-agent systems to carry out coordinated work in real-world applications. We have contributed a model of the new type of coordination knowledge as complex, coordination enhanced plans involving interactions by communicative action. The execution by agents of these plans results in multiple structured conversations taking place among agents. These ideas have been substantiated into a practical, application-independent coordination language that provides constructs for specifying the coordination-enhanced plans as well as the interpreter supporting their

Vol. 2(3), Apr, 2012, pp 58-75, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org
© Austrian E-Journals of Universal Scientific Organization

execution. The agent-based approach to system development offers a natural means of conceptualizing, designing and building distributed systems. The successful practice of this approach requires robust methodologies for agent-oriented software engineering. This paper applies MAS-CommonKADS, a methodology for agent-based system development, to the solution of a supply chain management system. MAS-CommonKADS is a knowledge engineering methodology; therefore, it can provide a good basis for MAS modeling since it deals with the development of knowledge based systems. The next step in this research is modeling the SCM system using two other methodologies, Tropos, and GAIA and comparing these three methodologies.

References

[1] C. Iglesias, M. Garijo, J. Centeno-Gonzalez, and V. J. R., "Analysis and Design of Multiagent Systems using MAS-CommonKADS," presented at Agent Theories, Architectures, and Languages, 1998.

[2] E. A. Arenas and G. Barrera-Sanabaria, "Applying the MAS-CommonKADS Methodology to the Flights Reservation Problem: Integrating Coordination and Expertise," presented at proceedings of The 5th Joint Conference on Knowledge-Based Software Engineerin, Maribor - Slovenia, 2002.

[3] SAP, "FINANCIAL SUPPLY CHAIN MANAGEMENT WITH SAP," SAP White Paper, Available:http://www.sap.com/solutions/businesssuite/erp/financials/pdf/BWP_WP_FSCM.pdf, 2005.

[4] M. Barbuceanu and M. S. Fox, "The Information Agent: An Infrastructure Agent Supporting Collaborative Enterprise Architectures," presented at Proceedings of Third Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, Morgantown, West Virginia, 1994.

[5] D. F. Pyke, "Supply Chain Management," C. Harris and S. Gass, Eds.: Encyclopedia of MS/OR, 2009.

[6] D. J. Thomas and P. M. Griffin., "Coordinated Supply Chain Management," European Journal of Operational Research, vol. 94, pp. 1-15, 1996.

[7] Y.Bassok and R.Akella., "Multiplan Coordination: Combined Production and Ordering with Demand and Supply Uncertainty," Management Science, vol. 37(12), pp. 15, 1991.

[8] J.Sun and N.M.Sadeh., "Coordinating Multi-Attribute Reverse Auctions Subject to Finite Capacity Considerations.," Carnegie Mellon University Technical Report, May 2003.

[9] M.Babaioff and N.Nissan., "Concurrent Auctions Across the Supply Chain," presented at Proceedings of ACM Conference on Electronic Commerce, Tampa, 2001.

[10] M.Babaioff and W.E.Walsh, "Incentive-Compatible, Budget-Balanced, yet Highly Efficient Auctions for Supply Chain Formation.," presented at Proceedings of the 4th ACM conference on Electronic commerce.

[11] W.E.Walsh and M.P.Wellman, "Decentralized Supply Chain Formation: A Market Protocol

[12] N.M.Sadeh, D. W. Hildum, D.Kjenstad, and A.Tseng, "MASCOT: An Agent Based Architecture for Dynamic Supply Chain Creation and Coordination in the Internet Economy," Journal of Production, Planning and Control, vol. 12(3), 2007.

[13] D.Zeng and K.Sycara., "Dynamic Supply Chain Structuring for Electronic Commerce Among Agents.In Intelligent Information Agents,," Chapter 10. Springer, 1999.

[14] J.Collins, W.Ketter, and M.Gini., "A Multi-Agent Negotiating Test bed for Contracting Tasks with Temporal and Precedence Constraints.," International Journal of Electronic Commerce, vol. 7(1), pp. 35-57, 2007.

[15] Y. Chen, Y. Peng, T. Finin, Y. Labrou, and S. Cost, "Negotiating agents for supply chain management," presented at Proceedings of the AAAI Workshop on Artificial Intelligence for Electronic Commerce, 1999.

[16] G. Schreiber, H. Akkermans, A. Anjewierden, R. deHoog, N. Shadbolt, W. VandeVeldere.

[17] J. Rumbaugh, M. Blaha, W. Premerlani, and V. F. Eddy, Object oriented modeling and design. Upper saddle river: NJ: Prentice Hall, 1991.

[18] I. Jacobson, M. Christerson, P. Jonsson, and G.Overgaard, Object-oriented software engineering: A use case driven approach. New York: ACM Press, 2008.

[19] R. Wirfs-Brock, B.Wilkerson, and L.Wiener, Designing object-oriented software. Upper saddle river: NJ: Prentice Hall, 2009.